**MAR BASELIOS COLLEGE OF ENGINEERING AND TECHNOLOGY**
***(Autonomous)***

**(Autonomous Institution under APJ Abdul Kalam Technological University)**

MAR BASELIOS
COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS

# Curriculum structure
# for
# Master of Technology
# In
# Artificial Intelligence and Data Science

(Year of Introduction: 2026)

Mar IvaniosVidyanagar
Nalanchira, Thiruvananthapuram,
Kerala, India. Pin: 695015

www.mbcet.ac.in

# PROGRAM OUTCOMES - PO

Program outcomes are the attributes that are expected to be demonstrated by a graduate after completing the course.

**PO1:** An ability to independently carry out research/ investigation and development work in engineering and allied streams

**PO2:** An ability to communicate effectively, write and present technical reports on complex engineering activities by interacting with the engineering fraternity and with society at large.

**PO3:** An ability to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program

**PO4:** An ability to apply stream knowledge to design or develop solutions for real world problems by following the standards

**PO5:** An ability to identify, select and apply appropriate techniques, resources and state-of-the-art tool to model, analyze and solve practical engineering problems.

**PO6:** An ability to engage in life-long learning for the design and development related to the stream related problems taking into consideration sustainability, societal, ethical and environmental aspects

**PO7:** An ability to develop cognitive load management skills related to project management and finance which focus on Entrepreneurship and Industry relevance.

The departments conducting the M. Tech course can define their own PSOs if required, and assessment shall also be done for the same.

# CONTENTS

# 1. Semester-wise Distribution of the Courses

**a)**

| Slot | Course Type | Course Code | Course | Marks | | Hours L - T - P | Credits |
|---|---|---|---|---|---|---|---|
| | | | | CIA | ESE | | |
| A | PCC | 26MA061B | Mathematical Foundations for Machine Learning and Data Science | 50 | 50 | 4 – 0 – 0 | 4 |
| B | PCC | 26CS261A | Foundations of Data Science | 50 | 50 | 4 – 0 – 0 | 4 |
| C | PCCP | 26CS261B | Foundations of Artificial Intelligence | 50 | 50 | 3 – 0 – 2 | 4 |
| D | PEC/ PECP | 26CS2XXX | Program Elective 1 | 50 | 50 | 3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 – 2 | 3 |
| E | PEC/ PECP | 26CS2XXX | Program Elective 2 | 50 | 50 | 3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 – 2 | 3 |
| S | AC | 26AC061A | Research Methodology & IPR | 50 | 50 | 2 – 0 – 0 | 0 |
| T | LBC | 26CS269A | Data Science Lab | 100 | - | 0 – 0 – 3 | 2 |
| | | | **Total** | **400** | **300** | **23 – 28** | **20** |

**b) Semester I (M1)Teaching Assistance: up to 6 hours**

**Program Elective Courses and Program Elective Courses with practical components in M1 and M2 must be chosen from a single basket of all PECs and PECPs.**

### c) Semester II (M2)

| Slot | Course Type | Course Code | Course | Marks | | Hours L - T – P | Credits |
|---|---|---|---|---|---|---|---|
| | | | | CIA | ESE | | |
| A | PCC | 26CS261C | Applied Machine Learning | 50 | 50 | 4 – 0 – 0 | 4 |
| B | PEC/ PECP | 26CS2XXX | Program Elective Course 3/ Program Elective Course with practical component 3 | 50 | 50 | 3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 – 2 | 3 |
| C | PEC/ PECP | 26CS2XXX | Program Elective Course4/ Program Elective Course with practical component 4 | 50 | 50 | 3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 – 2 | 3 |
| D | IEC/ SAEC* | 26CS2XXX | Industry Elective / (Skill/Ability Enhancement Course) | 50 | 50 | 3 – 0 – 0/ 2 – 1 – 0/ 2 – 0 – 2 | 3 |
| S | PR | 26CS267A | Mini project | 100 | - | 0 – 0 – 6 | 3 |
| T | LBC | 26CS269B | Applied Machine Learning Lab | 100 | - | 0 – 0 – 3 | 2 |
| | | **Total** | | **400** | **200** | **22- 26** | **18** |

**Teaching Assistance: Upto 6 hours**

*Marks / GPA earned in this SAEC will be used for awarding GPA for this course.

### d) Semester III (M3)

| Slot | Course Type | Course No | Course | Marks | | Hours L - T – P | Credits |
|---|---|---|---|---|---|---|---|
| | | | | CIA | ESE | | |
| A | SAEC** | | Industry Elective / (Skill/ Ability Enhancement Course) | To be successfully completed | | - | 3 |
| D | PR | 22CS278A | Project(Phase I)/ Project/ Internship | 100<br>100<br>100 | -<br>100<br>- | 0 – 0 – 24<br>0 – 0 – 24<br>Industry norms | 16 |
| | | **Total** | | **100** | **-/ 100/ -** | **24** | **19** |

**Teaching Assistance for students doing Project (Phase I)/ Project in the college: 5 hours**

** This SAEC can be carried out at any time from M1 to M3, and credited in M3.

### e) Semester IV (M4)

| Slot | Course Type | Course Code | Course | Marks | | Hours L - T – P | Credits |
|---|---|---|---|---|---|---|---|
| | | | | CIA | ESE | | |
| D | PR | 22CS278B | Project (Phase II)/<br>Project/<br>Internship | 100<br>100<br>100 | 100<br>100<br>- | 0 – 0 – 24<br>0 – 0 – 24<br>Industry norms | 16 |
| | | | **Total** | **100** | **100/100/ -** | **24/ Industry norms** | **16** |

**Teaching Assistance for students doing Project (Phase II)/ Project in the college: 5 hours**

## <u>Program Elective courses</u>

| # | Course code | Course Name |
|---|---|---|
| **1.** | 26CS262A | Data Analytics |
| **2.** | 26CS262B | Advanced programming in Python |
| **3.** | 26CS262C | Trustworthy Artificial Intelligence |
| **4.** | 26CS162K | High Performance Computing |
| **5.** | 26CS262E | Robotics |
| **6.** | 26CS162D | Deep Learning Techniques |
| **7.** | 26CS262G | Reinforcement Learning |
| **8.** | 26CS262H | Generative AI |
| **9.** | 26CS262I | Agentic AI |
| **10.** | 26CS262J | AI driven threat analysis for Cyber Security |
| **11.** | 26CS262K | Natural Language Processing |

**Industry Elective courses**

| # | Course code | Course Name |
|---|---|---|
| **1.** | 26CS266A | Cloud Computing for Artificial Intelligence |
| **2.** | 26CS266B | Foundations and Construction of Large Language Models |

# PROGRAM CORE COURSES (PCC)

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26MA061B | **MATHEMATICAL FOUNDATIONS FOR MACHINE LEARNING AND DATA SCIENCE** | PCC | 4 | 0 | 0 | **4** | 2026 |

## i) COURSE OBJECTIVES

This course introduces core mathematical foundations for data science, emphasizing linear algebra, probability, graph theory, and machine learning to model and analyze structured, networked data.

## ii) COURSE OUTCOMES

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

| | | |
|---|---|---|
| CO1 | Apply linear algebra techniques for data analysis, dimensionality reduction and machine learning models. | Apply |
| CO2 | Apply probability theory to model uncertainty in data-driven systems and machine learning systems. | Apply |
| CO3 | Apply statistical inference for data analysis, interpretation and model evaluation. | Apply |
| CO4 | Apply graph theory to model and solve networked data problems in machine learning. | Apply |
| CO5 | Apply graph algorithms to real-world connectivity, path and optimization problems. | Apply |

## iii) SYLLABUS

This course builds a strong mathematical foundation for data science by covering proof techniques, linear algebra, probability, and graph theory, with emphasis on rigorous reasoning and applications to structured data, networks, and computing systems.

## iv) REFERENCES

1) Gilbert Strang, Linear Algebra and Learning from Data, Wellesley-Cambridge Press.

2) Sheldon M. Ross, Introduction to Probability Models, Academic Press.

3) Casella, G. and Berger, R. L., Statistical Inference, Cengage Learning.

4) J. Truss, “Discrete Mathematics for Computer Scientists”, 2/e, Addison Wesley, 1999. Bernard Kolman, Robert C Busby, Sharon Kutler Ross, “Discrete Mathematical Structures”, 2/e, Prentice-Hall India Private Limited, 1996.

5) Ralph P. Grimaldi , B.V Ramana, Discrete and Combinatorial Mathematics, Fifth Edition, Pearson 2016

6) Doulgas B West, Introduction to Graph Theory, Prentice Hall India Ltd.

### (v) COURSE PLAN

| Module | Contents | No. of hours |
|---|---|---|
| I | Linear Algebra for Data Science and ML : Vector spaces, Matrix operations and properties, Eigenvalues, eigenvectors, diagonalization, Singular Value Decomposition (SVD) and low-rank approximation, Applications: Principal Component Analysis (PCA), dimensionality reduction. | 12 |
| II | Probability Theory and Random Variables : Axioms of probability, Conditional probability and Bayes' theorem, Random variables: discrete and continuous, Common probability distributions (Bernoulli, Binomial, Poisson, Gaussian, Exponential), Expectation, variance, covariance, correlation. | 14 |
| III | Statistical Inference and Estimation: Sampling methods, sampling distributions, Point estimation: Method of moments, Maximum Likelihood Estimation (MLE), Interval estimation, confidence intervals, Hypothesis testing and regression analysis | 11 |
| IV | Graph Theory Foundations for Data Science: Graph representations (adjacency, incidence matrices), Connectivity, Trees, spanning trees, Eulerian, Hamiltonian graphs, Planar graphs, Euler's formula (with proof) | 10 |
| V | Advanced Graph Algorithms and Applications: Graph traversal algorithms: BFS and DFS, Shortest path algorithms: Dijkstra, Bellman–Ford, Minimum spanning trees: Prim's, Kruskal's algorithms, Graph coloring, chromatic number, Applications of graph theory to network and ML applications. | 13 |
| | **Total hours** | **60** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS261A | **FOUNDATIONS OF DATASCIENCE** | PCC | 4 | 0 | 0 | **4** | 2026 |

**i) COURSE OBJECTIVES**

- To provide the fundamental knowledge on data science and to equip the students with knowledge, skills and tools necessary to thrive in today's data-driven world.
- Document and transfer the results and effectively communicate the findings using visualization techniques.
- This course serves as a bridge between theoretical foundations and real-world applications,offering the hands-on experience required to tackle actual data challenges.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| **CO1** | Apply Pandas functionalities to process structured datasets using real-world examples. | Apply |
| **CO2** | Apply advanced tools and techniques of Exploratory Data Analysis (EDA) to explore, summarize and visualize data. | Apply |
| **CO3** | Build data preprocessing pipelines using feature engineering to improve model performance. | Apply |
| **CO4** | Make use of advanced tools to develop Machine learning models | Apply |
| **CO5** | Develop models using Time series data to solve real world problems. | Apply |

**iii) SYLLABUS**

Introduction to Data Science, Python for Data Manipulation – Pandas Series and DataFrames, indexing and selection, data cleaning, aggregation, merging, reshaping, time-series handling, Advanced Data Wrangling and Feature Engineering – outlier detection, encoding categorical variables, scaling and transformations, feature construction and selection, handling imbalanced data, Regression and Model Evaluation – linear and logistic regression, regularization techniques, evaluation metrics, feature selection, model validation, Supervised and Unsupervised Learning – decision trees, random forests, boosting (XGBoost, LightGBM), SVM, clustering methods, dimensionality reduction, Time Series and Emerging Areas – ARIMA, Prophet, time-series forecasting, deep learning basics with Pandas, Pytorch, Advanced AI technologies

**iv) TEXTBOOKS**

1. Avrim Blum, John Hopcroft, Ravindran Kannan, "Foundations of Data Science", Cambridge University Press, 2020.

2. McKinney, Wes. Python for data analysis: Data wrangling with pandas, numpy, and jupyter. " O'Reilly Media, Inc.", 2022.

3. Zheng, Alice, and Amanda Casari. Feature engineering for machine learning: principles and techniques for data scientists. " O'Reilly Media, Inc.", 201.

4. Müller, Andreas C., and Sarah Guido. Introduction to machine learning with Python: a guide for data scientists. " O'Reilly Media, Inc.", 2016.

5. Nielsen, Aileen. Practical time series analysis: Prediction with statistics and machine learning. O'Reilly Media, 2019.

**REFERENCES**

1. Ani Adhikari and John DeNero, "Computational and Inferential Thinking: The Foundations of Data Science", GitBook, 2019.
2. Jake, VanderPlas. "Python Data Science Handbook.Essential Tools for Working with Data." (2016).
3. Géron, Aurélien. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. " O'Reilly Media, Inc", 2022.

**REFERENCE PAPERS**

**Data Science in Science, Volume 5, Issue 1 (2026)**

**Data Science in Science: Vol 5, No 1 (Current issue)**

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | **Basics of Data Science** -Introduction; Importance of linear algebra.**Python for Advanced Data Manipulation with Pandas: Getting Started with Pandas**: Creating Series and DataFrames, Viewing and inspecting data - **Basic Pandas Operations**: indexing and selection, boolean filtering, handling missing data, changing data types, renaming and replacing values, string operations, sorting data – **Data Transformation and Aggregation**: arithmetic operations between columns, applying functions, grouping data, using aggregation functions – **Advanced Pandas Operations**: merging and joining datasets, pivoting and reshaping, creating and working with MultiIndex DataFrames – **Working with Dates and Time**: parsing dates, extracting date parts, resampling time-series data. | 12 |
| II | **Advanced Data Wrangling and Feature Engineering** –**Outlier Detection and Treatment**: Identifying outliers using Interquartile Range (IQR), Z-score, and visualization techniques- Automated Exploratory Visualization using Pandas Profiling. Charts and visuals using web-based visualization tools- create Bar chart, Tool tips, Pie chart, Gauge chart, Funnel chart using Power BI and Tableau. Handling outliers with capping, flooring, or transformation methods. **Encoding Categorical Variables**: one-hot encoding, label encoding, ordinal encoding, frequency and target encoding, handling high-cardinality features effectively. | 12 |
| III | **Feature Transformation**: scaling techniques such as Standardization and Min-Max Scaling, normalization using L2 norm, transformations for skewed data using log, square root, Box-Cox, and Yeo-Johnson methods. **Feature Construction**: deriving new features from text, dates, and interaction terms, creating polynomial features to model non-linearity, feature binning. **Feature Selection Techniques**: filter methods (correlation analysis, mutual information), wrapper methods (Recursive Feature Elimination), and embedded methods (Lasso, Tree-based feature importance). **Handling Imbalanced Datasets**: identifying class imbalance, applying resampling methods such as Random Oversampling, SMOTE (Synthetic Minority Oversampling Technique), and under sampling techniques. | 14 |

| | | |
|---|---|---|
| IV | **Introduction to Machine Learning: Supervised and Unsupervised Learning Techniques Random Forests**: **Boosting Algorithms**: AdaBoost, Gradient Boosting, and advanced models like XGBoost and LightGBM, handling overfitting through early stopping and regularization – **Hyperparameter Tuning**: grid search, randomized search, and cross-validation techniques using GridSearchCV and RandomizedSearchCV. **Introduction to Deep Learning**: Basic implementation of Deep Learning models using Pandas, Pytorch, Hugging Face. | 10 |
| V | **Time Series Analysis and Introduction to Advanced AI technologies** – **Time Series Forecasting**: Understanding time series components, working with datetime indices, resampling, shifting, and rolling statistics – **Forecasting Models**: ARIMA and Seasonal ARIMA (SARIMA), model selection using AIC/BIC, exponential smoothing methods, and Prophet for robust forecasting – **Model Diagnostics**: ACF and PACF plots, stationarity tests (ADF, KPSS), residual analysis. | 12 |
| | **Total Hours** | **60** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS261B | **FOUNDATIONS OF ARTIFICIAL INTELLIGENCE** | PCCP | 3 | 0 | 2 | **4** | 2026 |

**i) COURSE OBJECTIVES**

The course introduces the variety of concepts in the field of artificial intelligence. It discusses the philosophy of AI, and how to model a new problem as an AI problem. The course prepares a student to take a variety of focused, advanced courses in various subfields of AI.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| **CO1** | Explain AI problem characteristics, intelligent agent models, and environment types for real-world applications. | Understand |
| **CO2** | Apply uninformed and informed search strategies, constraint satisfaction techniques, and game playing algorithms to solve complex AI problems. | Apply |
| **CO3** | Apply logical representation techniques and inference mechanisms to construct knowledge-based systems. | Apply |
| **CO4** | Summarize probabilistic reasoning models and decision-making frameworks under uncertainty. | Understand |
| **CO5** | Outline deep learning and generative AI models with consideration of ethical and responsible AI principles. | Understand |

**iii) SYLLABUS**

History and evolution, Intelligent agents. Uninformed search, Breadth First Search, Depth First Search, Uniform Cost Search, Informed search, Greedy Best First Search, A* algorithm, Heuristic functions, Local search algorithms, Constraint Satisfaction Problems, Backtracking search, Game playing, Minimax algorithm, Alpha-Beta pruning. Knowledge representation, Propositional logic, First Order Logic, Syntax and semantics of FOL. Uncertainty in AI, Sources of uncertainty, Probability theory basics, Conditional probability, Bayes theorem, Bayesian networks, Inference in Bayesian networks, Hidden Markov Models. Advanced AI Techniques: Deep learning overview, Generative AI fundamentals, Generative models, Autoencoders, Variational Autoencoders, Generative Adversarial Networks, Transformers architecture overview.

**iv)TEXTBOOKS**

1. Stuart Russell & Peter Norvig, Artificial Intelligence: A Modern Approach, Prentice-Hall, Third Edition, 2009.

2. Ian Good Fellow, Yoshua Bengio & Aaron Courville, Deep Learning, MIT Press, 2016.

**v) COURSE PLAN**

| **Module** | **Contents** | **No. of hours** |
|---|---|---|
| I | **Introduction to AI:** History and evolution, Intelligent agents – Types, PEAS description, Problem formulation and state space representation, AI problem characteristics, Performance measures and environment types, Applications of AI in various domains | 8 |
| II | **Search Strategies and Problem Solving:** Uninformed search, Breadth First Search, Depth First Search, Uniform Cost Search, Informed search, Greedy Best First Search, A* algorithm, Heuristic functions, Local search algorithms, Constraint Satisfaction Problems, Backtracking search, Game playing, Minimax algorithm, Alpha-Beta pruning | 10 |
| III | **Knowledge Representation and Reasoning**: Knowledge representation, Propositional logic, First Order Logic, Syntax and semantics of FOL, Inference mechanisms, Resolution, Rule-based systems, Semantic networks, Frames, Ontologies, Forward chaining, Backward chaining | 10 |

| | | |
|---|---|---|
| IV | **Uncertainty and Probabilistic Reasoning**: Uncertainty in AI, Sources of uncertainty, Probability theory basics, Conditional probability, Bayes theorem, Bayesian networks, Inference in Bayesian networks, Hidden Markov Models, Temporal probabilistic models, Decision theory, Utility theory, Expected utility principle, Decision networks, Markov Decision Processes, Handling incomplete and noisy data | 10 |
| V | **Advanced AI Techniques:** Deep learning overview, Generative AI fundamentals, Generative models, Autoencoders, Variational Autoencoders, Generative Adversarial Networks, Transformers architecture overview, Large Language Models, Prompt engineering basics, Foundation models, Multimodal AI systems, Ethical issues in AI, Explainable AI, Responsible AI, Recent trends in AI research | 7 |
| | **Total Hours** | **45** |

**LAB/PRACTICAL**

| Sl. No. | Experiment | Hours |
|---|---|---|
| 1 | Write a program to formulate and solve the 8-Puzzle problem using state space representation. | 3 |
| 2 | Implement Breadth First Search (BFS) and Depth First Search (DFS) algorithms for a given graph and compare their performance. | 3 |
| 3 | Develop a program to implement A* Search algorithm using an appropriate heuristic function. | 3 |
| 4 | Write a program to solve the N-Queens problem using Backtracking (Constraint Satisfaction approach). | 3 |
| 5 | Implement the Minimax algorithm for a two-player game (e.g., Tic-Tac-Toe). | 3 |
| 6 | Implement the Minimax algorithm using Alpha-Beta pruning and analyze the reduction in search space. | 3 |
| 7 | Develop a program to represent knowledge using Propositional Logic and perform inference using truth table or resolution method. | 3 |

| | | |
|---|---|---|
| 8 | Implement Forward Chaining and Backward Chaining for a simple rule-based system. | 3 |
| 9 | Write a program to perform probabilistic inference using Bayes Theorem and construct a simple Bayesian Network. | 3 |
| 10 | Implement a basic Generative AI model (e.g., Autoencoder or simple GAN) and analyze its output. | 3 |
| | **Total Hours** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS261C | **APPLIED MACHINE LEARNING** | PCC | 4 | 0 | 0 | **4** | 2026 |

**i) COURSE OBJECTIVES**

This course introduces the basic concepts of machine learning. The concepts covered in the course include Regression, Classification, Clustering and Optimization techniques. This course helps the students to develop solutions to real world applications using machine learning techniques.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Illustrate machine Learning concepts, basic parameter estimation methods and regression techniques. | Understand |
| CO2 | Apply various classification algorithms to build predictive models on real-world datasets. | Apply |
| CO3 | Demonstrate the working of clustering algorithms and neural networks for typical machine learning applications. | Apply |
| CO4 | Apply model evaluation and validation techniques to assess and improve machine learning models. | Apply |
| CO5 | Demonstrate various optimization algorithms in machine learning problems. | Apply |

**iii) SYLLABUS**

Feature Engineering, Parameter Estimation, Regression, Classification (Margin-based, Probability-based, Decision Tree, Ensemble tree, Instance-based), Clustering (Density based, Distribution based), Neural Networks, Evaluation techniques and Optimization .

**iv) TEXTBOOKS**

1. Shai Shalev-Shwartz and Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2017.
2. Sergios Theodorodos, Machine Learning, 3rd Edition, Elsevier, 2025.
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 1st Edition, MIT Press, 2016.

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Introduction—Introduction to Machine Learning, AI vs. ML, Types of learning, Types of Data (Tabular, Image, Video, Audio, Sequential), Feature Engineering<br>Training and test sets, Concept of over fitting, under fitting, Bias and Variance.<br>Basics of parameter estimation - maximum likelihood estimation (MLE) and maximum a posteriori estimation (MAP).<br>ML approaches - Introduction to regression, classification, clustering. Regression - Linear Regression, Multiple Linear Regression, Support Vector Regression, Ridge regression, Lasso Regression | 13 |
| II | Classification—Probabilistic Classifiers (Naïve Bayes, Logistic regression),Instance-Based Learning (K-Nearest Neighbors),Margin-Based Classifiers (Support Vector Machine), Decision Trees (Entropy, Information Gain, Tree construction, ID3, Issues in Decision Tree Learning), Ensemble Trees (Random Forest), and Classification by Regression (CART). | 13 |
| III | Neural Networks - The Perceptron, Activation Functions, Training Feed Forward Network by Back Propagation.<br>Evaluation: Train-test split, Cross-validation, k-fold validation, Stratified k-fold validation, Bootstrapping, Cross-entropy loss, Binary cross-entropy, Regularization, Dropouts, Confusion matrix, AUC-ROC, EER, RMS, Precision, Recall and mAP. | 12 |
| IV | Clustering—Density-based, Distribution-based, Kmeans, DBSCAN, Hierarchical Clustering, Assessment Metrics for Clustering Algorithms.<br>Dimensionality reduction—PCA. | 11 |
| V | Optimization Techniques—Gradient-Based Methods (Gradient Descent, Nesterov Accelerated Gradient, Adagrad, RMSProp), Derivative-Free Methods (Grid Search, Random Search), Metaheuristic Methods (Genetic Algorithm, Particle Swarm Algorithm, Ant Colony Optimization) | 11 |
| | **Total hours** | **60** |

# LABORATORY COURSES (LBC)

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS269A | **DATA SCIENCE LAB** | LBC | 0 | 0 | 3 | **2** | 2026 |

**i) COURSE OBJECTIVES**

The Applied Data Science Lab aims to provide practical skills in data preprocessing, feature engineering, machine learning model building, optimization, and evaluation using Python. It also introduces students to clustering, recommendation systems, and time-series forecasting, helping them implement an end-to-end real-world data science project.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | **Make Use** of appropriate preprocessing techniques to clean real-world datasets | Apply |
| CO2 | Apply **feature engineering**, and **feature scaling/transformations** to prepare data for modelling. | Apply |
| CO3 | Develop and evaluate supervised learning models to improve model generalization using regularization techniques | Apply |
| CO4 | Build and optimize advanced machine learning solutions using **boosting algorithms**, handle **imbalanced dataset** | Apply |
| CO5 | Implement **unsupervised learning and time-series analytics** | Apply |

**iii) COURSE PLAN**

| Experiments | Contents | No. of hours |
|---|---|---|
| I | Develop Automated Exploratory Visualization using Pandas Profiling | 3 |
| II | Create traditional and advanced charts using Power BI and Tableau | 3 |
| III | Apply different encoding methods and compare results. | 3 |
| IV | Apply Feature Scaling and Transformation for Skewed Data Normalize and transform features for better model performance. | 3 |
| V | Handle Imbalanced Dataset using SMOTE and Sampling | 3 |
| VI | Build a Linear Regression model and evaluate performance and Reduce overfitting using regularization techniques. | 3 |
| VII | Build advanced boosting models with early stopping and regularization | 4 |
| VIII | Optimize model parameters using tuning techniques- Tuning using GridSearchCV and RandomizedSearch | 3 |
| IX | Build clustering using DBSCAN and detect noise/outliers. | 3 |

| | | |
|---|---|---|
| X | Make use of Numpy, Pytorch, Hugging Face to develop Deep Learning models | 3 |
| XI | Recommend items using user-item interaction | 4 |
| XII | Build ARIMA model and forecast future values | 4 |
| XIII | Use ACF/PACF plots to identify ARIMA parameters | 3 |
| XIV | Build a complete forecasting pipeline | 3 |
| | **Total hours** | **45** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS269B | **MACHINE LEARNING LAB** | LBC | 0 | 0 | 3 | **2** | 2026 |

**COURSE OVERVIEW**:

This course enables the learners to get hands-on experience in most popular supervised learning algorithms (such as linear regression, logistic regression, decision trees, Bayesian learning and Naive Bayes algorithm) and unsupervised learning algorithms (such as basic clustering algorithms). This helps the learners to understand the process of knowledge inference from raw data through dataset preprocessing and analysis.

.

## COURSE OUTCOMES

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

| | | |
|---|---|---|
| CO 1 | Implement machine learning algorithms using packages and libraries in Python for various applications | Apply |
| CO 2 | Implement python programs for supervised learning methods through Neural network, Regression, and classification | Apply |
| CO 3 | Implement clustering algorithms. | Apply |
| CO 4 | Apply dimensionality reduction as a dataset preprocessing step. | Apply |

## SYLLABUS

Familiarization using Pandas, Numpy and Visualization tools, Decision Trees, Classifier Performance, kNN, Multi-class Classification, Naive Bayes, Support Vector Machines, Neural Networks, Clustering, Dimensionality Reduction Techniques

## COURSE PLAN

| SL No. | Topics | No. of hours |
|---|---|---|
| 1 | Pandas, Numpy and Visualization tools | 9 |
| 2 | Implement K-Nearest Neighbor algorithm to classify any dataset | 3 |
| 3 | Implement and demonstrate Single, Multi variable and Polynomial Regression for a given set of training data stored in a .CSV file and evaluate the accuracy. | 3 |
| 4 | Implement a Python program to perform logistic regression on a dataset. | 3 |

| | | |
|---|---|---|
| 5 | Write a Python program to implement Naive Bayes classifier and calculate the accuracy, precision, and recall for your data set. | 3 |
| 6 | Write a Python program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample. | 3 |
| 7 | Assuming a set of data that need to be classified, use a Support Vector Machine classifier to perform this task and evaluate the accuracy. | 3 |
| 8 | Implement dimensionality reduction using PCA. | 3 |
| 9 | Implement K-Means Clustering using any given dataset. | 3 |
| 10 | Implement Agglomerative Hierarchical Clustering. | 3 |
| 11 | Build an Artificial Neural Network using Backpropagation algorithm and test the same with appropriate dataset. | 4 |
| 12 | Implement different optimization techniques for training a neural network and compare their convergence behavior and accuracy | 5 |
| | **Total** | **45** |

# PROGRAM ELECTIVE COURSES (PEC)

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262A | **DATA ANALYTICS** | PEC | 2 | 0 | 2 | **3** | 2026 |

**i) COURSE OBJECTIVES**

This course is offered to introduce fundamental algorithmic ideas in processing data. The preliminary concepts of Hadoop and Map Reduce are included as part of this course.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| **CO1** | Apply data science concepts to identify real-world problems | Apply |
| **CO2** | Apply big data concepts by demonstrating Hadoop basics, HDFS operations, and MapReduce processing with YARN. | Apply |
| **CO3** | Apply Hadoop MapReduce concepts to analyze data . | Apply |
| **CO4** | Apply Power BI tools to import, clean, model, and analyze data using Power Query and DAX | Apply |
| **CO5** | Apply MLOps practices to build and deploy reproducible machine learning workflows. | Apply |

**iii)SYLLABUS**

Big data fundamentals, five V's, analytics practices, real-world use cases, and overview of Apache Hadoop and its ecosystem.

HDFS architecture, design principles, daemons, metadata management, and data read/write operations.

MapReduce framework including stages, job anatomy, scheduling, shuffle and sort, task execution, and YARN.

Big data management tools: Pig (Pig Latin, execution modes), Hive (architecture, HiveQL), and introductory NoSQL concepts.

**iv)TEXTBOOKS**


1. Davy Cielen, Arno D. B. Meysman, and Mohamed Ali ,"Introducing Data Science - Big data, machine learning, and more, using Python tools" , Dreamtech Press 2016.
2. Michael Minelli, Michelle Chambers, and AmbigaDhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses",Wiley,2013
3. EMC Education Services, "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", Wiley ,January 2015
4. Tom White,"Hadoop: The Definitive Guide", Third Edition, O'Reilley,2012.
5. Eric Sammer,"Hadoop Operations",O'Reilly Media, Inc ,2012

**6.** E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012. 7. "Programming Pig", Alan Gates, O'Reilley,2011.

**REFERENCES**


- Sourabh Mukherjee, Amit Kumar Das and Sayan Goswami, " Big Data Simplified", Pearson, 1st edition, 2019.
- Murtaza Haider, "Getting Started with Data Science", First Edition, Kindle Edition, IBM Press, 2015.
- Thomas Erl, Wajid Khattak and Paul Buhler " Big Data Fundamentals:Concepts, Drivers and Techniques", Prentice Hall, Pearson Service, 2016.


**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| [ I | Data Analytics Types of Analytics Data Science vs Data Analytics vs Big Data, Data Understanding, Structured / Semi-structured / Unstructured data Parquet Data Quality & Data Governance Basics, Sampling and Data Collection Strategies, Intro to Data Storage Concepts, Basic Statistical Concepts for Analytics, Introduction to EDA Tools, Feature Engineering Basics, Model Evaluation Concepts | 6 |
| II | Big Data Overview–the five V's of big data-State of the Practice in Analytics, Examples of Big Data Analytics-Apache Hadoop and the Hadoop Ecosystem-HDFS-Design of HDFS, HDFS Concepts-Daemons-Reading and Writing Data-Managing File system Metadata- Map Reduce-The Stages of Map Reduce -Introducing Hadoop Map Reduce ,Daemons-YARN | 6 |
| III | Analyzing the Data with Hadoop using Map and Reduce-Developing a Map Reduce Application-Anatomy of a Map Reduce Job-Scheduling-Shuffle and Sort - Task execution. Big data Management Tools: PIG- : Introduction to PIG, Execution Modes of Pig, Pig Latin, HIVE: Hive Architecture, HIVEQL, Introduction to NoSQL. | 6 |
| IV | Power BI overview and workflow (Desktop, Service, reports, dashboards),Data import from Excel/CSV/SQL and refresh basics ,Data cleaning using Power Query (filter, split, merge, append, missing values), Data modeling (relationships, star schema, columns vs measures), DAX basics for calculations and KPIs (CALCULATE, filters, time intelligence), Creating visuals and interactive reports (charts, slicers, drill-down, formatting)<br>Publishing and sharing in Power BI Service (dashboards, workspaces) | 6 |

| | | |
|---|---|---|
| V | Introduction to MLOps (lifecycle, MLOps vs DevOps/DataOps)<br>ML workflow recap (training, evaluation, reproducibility)<br>Version control for code, data, and models (Git, DVC)<br>Experiment tracking & model registry (MLflow)<br>Automated ML pipelines and CI/CD (GitHub Actions/Jenkins)<br>Model packaging & environment management (Docker)<br>Deployment & model serving (API, batch, real-time)<br>Scaling basics (Kubernetes intro)<br>Governance, security, compliance, Responsible AI<br>Cloud deployment basics (AWS/Azure/GCP) | 6 |
| | **Total Hours** | **30** |

**LAB / PRACTICAL**

| | **Experiment** | **No. of hours** |
|---|---|---|
| **Module-1** | | |
| **1** | Module 1: Introduction to Big Data & Ecosystem (Setup & Acquisition)<br><br>a) Environment Setup & HDFS Management: Install and configure Apache Hadoop in Pseudo-Distributed mode. Practice Hadoop Distributed File System (HDFS) commands (copyFromLocal, copyToLocal, mkdir, ls, rm) to manage data.<br><br>b) Big Data Ingestion (Informatica/Sqoop): Utilize Apache Sqoop to import data from a relational database (MySQL/Oracle) into HDFS, or use a data integration tool (like Pentaho Data Integration - Kettle) to ingest raw data.<br><br>c)Security & Auditing: Configure HDFS permissions and run auditing logs to track file access. | **6** |
| **Module-2** | | |

| | | |
|---|---|---|
| **2** | Module 2: Data Analysis & Processing (MapReduce & Hive)<br>a)Word Count MapReduce: Implement a Java-based MapReduce program to perform a word count on a large text file to understand the "4 Vs" (Volume, Variety).<br><br>b) Advanced MapReduce: Write a MapReduce program to analyze weather data (find max/min temperature) or perform Matrix Multiplication to process structured/semi-structured data.<br><br>c) Data Analysis with Hive: Create Hive tables (Internal/External), perform data loading, and execute HiveQL queries to analyze dataset patterns. | **6** |
| Module-3 | | |
| **3** | Module 3: Data Analysis Tools & Visualization (Business Analytics)<br>a)Pentaho Data Integration (PDI): Use Pentaho (Spoon) to design transformations and jobs for cleaning, filtering, and aggregating data from disparate sources.<br><br>b) Business Reporting (Cognos/Microstrategy): Connect to a data source (Hive or MySQL) and create interactive dashboards using Cognos Analytics or MicroStrategy to visualize data trends. | **6** |

| Module-4 | | |
|---|---|---|
| **4** | Module 4: Intelligent Predictive Analytics & Machine Learning<br><br>a) Unsupervised Learning (K-Means Clustering): Implement K-Means clustering in Hadoop/Spark to group similar data points (e.g., customer segmentation).<br><br>b) Association Rule Mining (Apriori Algorithm): Implement the Apriori algorithm on a market basket dataset to find frequently occurring itemsets.<br><br>· c)Supervised Learning (Neural Networks): Use a Python library (like Scikit-learn or TensorFlow) to implement a basic neural network or Kohonen model on a large dataset. | **6** |
| Module-5 | | |
| **5** | Module 5: Stream Computing & Real-time Analytics<br>a) Data Streaming Basics: Setup a stream processing environment and implement data sampling techniques to analyze streaming data in real-time.<br><br>b) Real-time Analytics (IBM InfoSphere Streams): Develop a simple application using IBM InfoSphere Streams (or Spark Streaming) to | **6** |

| | | |
|---|---|---|
| | process streaming data and estimate moments in a stream. | |
| | **Total Hours** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262G | **REINFORCEMENT LEARNING** | PECP | 2 | 0 | 2 | **3** | 2026 |

**i) COURSE OBJECTIVES**

- To understand the fundamentals of Reinforcement Learning.
- To learn about different types of Reinforcement Learning agents.
- To implement the various Reinforcement Learning algorithms
- To apply the Reinforcement Learning techniques to solve real-world problems.
  These objectives aim to equip students with the necessary skills and knowledge to effectively utilize Reinforcement Learning(RL) techniques in various real time applications.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Explain the fundamental concepts and elements of Reinforcement Learning | Understand |
| CO2 | Apply Markov Decision Process to model the decision-making problems in RL. | Apply |
| CO3 | Apply On policy and Off policy methods to solve the RL problems | Apply |
| CO4 | Apply value function approximation, n-step Sarsa, and TD($\lambda$) to solve the uncertainty problems. | Apply |
| CO5 | Make use of policy-based reinforcement learning algorithms to solve control problems. | Apply |

**iii) SYLLABUS**

Introduction to RL, Examples, Elements of RL, Multi-armed bandit problems, Markov Decision Process, Goals and rewards, Returns and episodes, Policies and value functions, Monte Carlo prediction, estimation and control, TD prediction, Sarsa, Q-learning, Value function approximation, Stochastic gradient methods, Linear methods, Non-linear function approximation, Policy approximation, Policy gradient theorem, REINFORCE algorithm, Actor-Critic methods, Trust-Region Policy Optimization, Proximal Policy Optimization, Introduction to OpenAI Gym.

**iv) TEXT BOOK**

1. Reinforcement learning: An introduction, Richard S. Sutton and Andrew G. Barto, Second edition, MIT Press, 2018.

**References**

Bilgin, E. (2020). Mastering reinforcement learning with Python: Build next-generation, self-learning models using reinforcement learning techniques and best practices (1st ed.). Packt Publishing.

Winder, P. (2020). *Reinforcement learning: Industrial applications of intelligent agents*. O'Reilly Media.

Zai, A., & Brown, B. (2020). *Deep reinforcement learning in action* (1st ed.). Manning Publications.

**Reference Papers**

1. John Schulman, Sergey Levine, Philipp Moritz, Michael Jordan, and Pieter Abbeel. Trust region policy optimization. In Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37 (ICML'15).
2. Schulman, John, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg limov. "Proximal policy optimization algorithms." arXiv preprint arXiv:1707.06347 (2017).

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Introduction to RL, Examples, Elements of RL, Multi-armed bandit problems, Action value methods | 4 |
| II | Markov Decision Process, Goals and rewards, Returns and episodes, Policies and value functions, Policy evaluation, Policy improvement, Policy iteration, Value iteration. Monte Carlo prediction, estimation and control. | 6 |
| III | Off Policy and On policy Methods- Sarsa, Q-learning, n-step TD prediction, n-step Sarsa, n-step Off-policy learning. | 7 |
| IV | Value function approximation, Stochastic gradient methods, Linear methods, Non-linear function approximation, Episodic semi-gradient control, Semi-gradient n-step Sarsa, TD(λ). | 7 |
| V | Policy approximation, Policy gradient theorem, REINFORCE algorithm, Actor-Critic methods, Proximal Policy Optimization. | 6 |
| | **Total Hours** | **30** |

**LAB / PRACTICAL**

| **Modules** | **Experiment No** | **Experiments** | **No. of hours** |
|---|---|---|---|
| **I** | **1** | To implement Multi-Armed Bandits (Epsilon-Greedy and UCB) by introducing the simplest RL setup where an agent chooses from multiple slot machines to maximize reward. | **4** |
| **II** | **2** | To implement the Monte Carlo Algorithm for modelling a basic Reinforcement Learning system. | **4** |
| **III** | **3** | To implement Tabular Q-Learning in Grid world. Implement Q Learning in a discrete grid environment where the agent must reach a goal state. | **4** |
| **III** | **4** | To implement SARSA in a Maze Navigation Task. Compare SARSA with Q-Learning in a grid-based maze. | **4** |
| **IV** | **5** | To implement Policy Gradients (REINFORCE) on a Simple Continuous Task. | **4** |
| **V** | **6** | To implement and train an agent with Actor-Critic Method. | **4** |
| **V** | **7** | Case study implementation using Open-Gym Library | **6** |
| | | **Total Hours** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS162K | **HIGH PERFORMANCE COMPUTING** | PEC | 3 | 0 | 0 | **3** | 2026 |

**i) COURSE OBJECTIVES:**

This course helps the learners to understand the different architectural features of high-end processors. This course discusses the Basics of high-end processors Architecture, Instruction-Level Parallelism, Data-Level Parallelism, Thread Level Parallelism, and GPU Architectures. This course enables the students to provide solutions to real-world problems Making use of the capabilities of HPC systems.

**ii) COURSE OUTCOMES:**

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

| | | |
|---|---|---|
| CO1 | Choose different types of modern processing environments and parallel computing hardware | Apply |
| CO2 | Make use of the concepts of Instruction Level Parallelism and thread level parallelism in branch prediction and multithreading. | Apply |
| CO3 | Apply the idea of Data Level Parallelism | Apply |
| CO4 | Apply the concept of shared memory architectures. | Apply |
| CO5 | Identify the advanced features of GPU architecture. | Apply |

**iii) SYLLABUS**

Classes of Computers, Parallelism and Parallel Architectures, Computer Architecture Fundamentals, Dependability and Quantitative Design Principles, Memory Hierarchies and Virtual Memory, Pipelining, Instruction-Level Parallelism and Compiler Techniques, Branch Prediction and Hardware Speculation, Multithreading and Thread-Level Parallelism, Vector Architectures and SIMD Extensions, Graphics Processing Units and Loop-Level Parallelism, Multiprocessor Architectures, Shared and Distributed Memory Systems, Cache Coherence and Synchronization, Memory Consistency Models, CPU–GPU Systems and Accelerated Computing Platforms, GPU Architecture and Memory Spaces, CPU–GPU Data Transfer and PCI Bus, Multi-GPU Platforms and Performance Benefits.

**iv) REFERENCES:**

1. John L. Hennessy, David A. Patterson Computer Architecture, A Quantitative Approach, Morgan Kaufman, Seventh Edition,2025.

2. Robert Robey, Yuliana Zamora, Parallel and High-Performance Computing, Manning Publications, First Edition, 2021.

3. Thomas Sterling, Matthew Anderson, and Maciej Brodowicz, High-Performance Computing–Modern Systems and Practices, First Edition,2017.

4. Charles Severance, Kevin Dowd, High-Performance Computing, O'Reilly Media, Second Edition,1998.

5. Kai Hwang, Faye Alaye Briggs, Computer Architecture and Parallel Processing, McGraw-Hill,1984.

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Classes of Computers- Classes of Parallelism and Parallel Architectures– Defining Computer Architecture– Dependability– Quantitative Principles of Computer Design– Basics of Memory Hierarchies– Virtual Memory and Virtual Machines– Pipelining | 7 |
| II | Instruction-Level Parallelism: Concepts and Challenges – Basic Compiler Techniques for Exposing ILP – Reducing Branch Costs With Advanced Branch Prediction – Hardware- Based Speculation – Multithreading: Exploiting Thread-Level Parallelism to Improve Uniprocessor Throughput | 9 |
| III | Vector Architecture – SIMD Instruction Set Extensions for Multimedia – Graphics Processing Units – Detecting and Enhancing Loop-Level Parallelism | 10 |
| IV | Multiprocessor Architecture: Issues and Approach – Centralized Shared-Memory Architectures– Performance of Symmetric Shared-Memory Multiprocessors– Distributed Shared-Memory and Directory-Based Coherence – Synchronization: The Basics – Introduction to Memory Consistency | 9 |
| V | The CPU-GPU system as an accelerated computational platform – The GPU and the thread engine – Characteristics of GPU memory spaces – The PCI bus: CPU to GPU data transfer overhead – Multi-GPU platforms – Potential benefits of GPU – accelerated platforms | 10 |
| | **Total hours** | **45** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262B | **ADVANCED PROGRAMMING IN PYTHON** | PECP | 2 | 0 | 2 | **3** | 2026 |

Pre-requisite: Basic knowledge in Python programming fundamentals

**i) COURSE OBJECTIVES**

1) To apply advanced Python programming concepts to design efficient solutions for industry-standard problems.
2) To perform advanced data preprocessing tasks, including data merging and data munging, for real-world datasets.
3) To design and develop robust and scalable web applications using Python.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Apply advanced Python programming concepts to develop efficient and modular programs. | Apply |
| CO2 | Apply data processing and analysis techniques using NumPy and Pandas on real-world datasets. | Apply |
| CO3 | Apply data wrangling and preprocessing methods to solve practical case studies. | Apply |
| CO4 | Apply web development skills using frameworks such as Django, Flask, FastAPI, and web2py with concurrency concepts. | Apply |
| CO5 | Apply database, IoT, and predictive modeling techniques to develop modern web applications. | Apply |

**iii)SYLLABUS**

Python data structures (lists, tuples, dictionaries, sets), object-oriented programming concepts including classes, objects, inheritance, and polymorphism, and multithreading and multiprocessing techniques. Data processing and analysis using CSV, Excel, JSON, NumPy, and Pandas, including data wrangling, merging, and munging, along with data visualization using Matplotlib. Data Science perspectives, such as working with Series and DataFrames, grouping, aggregation, and metrics creation.Deploy web applications using frameworks like Django, Flask, FastAPI, and web2py, integrating concurrency concepts such as multithreading and multiprocessing. Web application development using Web2Py, database programming with SQL and NoSQL, and building predictive models integrated with IoT devices. Recent trends in Python, Data Science, Web Applications, and IoT.

**iv)TEXTBOOKS**

1. Fluent Python, 2nd Edition – Luciano Ramalho(O'Reilly Media, **2022**) (Advanced Python concepts, idiomatic usage, data structures, concurrency)

2. Python for Data Analysis (3rd Edition) – Wes McKinney(O'Reilly Media, **2022**) (Data manipulation using pandas, NumPy, and Jupyter)

3. Build Python Web Apps with Streamlit – Aneev Kochakadan(Apress, **2025**) (Practical guide to building Python web applications).

### v) REFERENCES

1. Ani Adhikari and John DeNero, "Computational and Inferential Thinking: The Foundations of Data Science", GitBook, 2019.

2. Cathy O"NeiI and Rachel Schutt, "Doing Data Science: Straight Talk from the Frontline", O"ReiIIy Media, 2013.

### v) COURSE PLAN

| Module | Contents | No. of hours |
|---|---|---|
| I | **Module 1: Advanced Python Programming**<br>**Python data structures:** Lists, Tuples, Dictionaries, Sets; Functions, exceptions, lambda functions, map and filter operations; Itertools, generators, parallel processing concepts; Object-oriented programming: Classes, objects, data abstraction, encapsulation, inheritance, polymorphism, modularity, and code organization. | 7 |
| II | **Module 2: Data Processing and Analysis**<br>Handling CSV, Excel, JSON data; NumPy arrays: indexing, slicing, multidimensional arrays, array manipulation; Pandas DataFrames and Series, grouping, aggregation, merging, and summary tables; Metrics creation for analysis and Date and Time Manipulation. | 5 |
| III | **Module 3: Data Handling Techniques and Case Studies**<br>Data wrangling, merging, joining, and munging; Case study: Loan Prediction problem; Preparing real-world datasets for analysis. | 5 |
| IV | **Module 4: Python Web Frameworks and Concurrency**<br>**Introduction to Web Frameworks** (Overview and Comparison): Overview of Web Application Architecture, Introduction to Django, Introduction to Flask, Introduction to FastAPI, Comparison of Django, Flask, and FastAPI, Building a simple end-to-end mini application (basic routing, API creation, simple frontend template). **Basics of multithreading and multiprocessing in Python;**<br>Threading module and concurrent execution; Multithreaded priority queue and performance considerations. | 8 |
| V | **Module 5: Web Applications, IoT, and Recent Trends**<br>Web application development using Web2Py; Database programming with SQL and NoSQL;<br>Embedded applications with IoT devices; Predictive model development for IoT-based web apps; Recent trends in Python , Data Science , Web Applications and IoT. | 5 |

| | | | |
|---|---|---|---|
| | | **Total Hours** | **30** |

**LAB / PRACTICAL**

| | Experiments | No. of hours |
|---|---|---|
| **Module-1** | | |
| 1 | a) Store employee/student data using lists, tuples, dictionaries, and sets.<br>b) Write functions to compute summary statistics.<br>c) Use lambda, map, and filter to extract specific records.<br>d) Implement exception handling for invalid inputs.<br>e) Create a simple class-based design demonstrating:<br>• Encapsulation<br>• Inheritance<br>• Polymorphism | 2 |
| **Module-2** | | |
| 2 | a) Load data from CSV / Excel / JSON.<br>b) Perform NumPy operations: indexing, slicing, reshaping.<br>c) Use Pandas DataFrames for:<br>• Grouping and aggregation<br>• Merging datasets<br>d) Generate summary tables and key metrics.<br>e) Perform date-based analysis. | 4 |
| **Module-3** | | |
| 3 | a) Handle missing values and outliers.<br>b) Perform data wrangling and munging.<br>c) Apply encoding and normalization. | 2 |
| **Module-4** | | |
| 4 | Develop a small end-to-end web application using Django / Flask / FastAPI including: Core logic implementation, REST API development, and Basic frontend integration. | 4 |

| | | |
|---|---|---|
| 5 | • Read multiple data files concurrently using multithreading.<br>• Implement a thread-safe priority queue for task scheduling.<br>• Measure execution time for:<br>• Sequential execution<br>• Multithreaded execution<br>• Perform a CPU-intensive task using multiprocessing. | 4 |
| **Module-5** | | |
| 5 | • Develop a basic Web2Py web app.<br>• Store sensor data using SQL or NoSQL.<br>• Simulate IoT sensor readings.<br>• Display data on a web dashboard. | 4 |
| **Project** | | |
| 7 | Integrate all concepts learned across modules to do a project which includes the following:<br>a) Collect and store data from IoT sensors.<br>b) Use multithreading for data ingestion.<br>c) Perform data analysis using Pandas and NumPy.<br>d) Apply data cleaning techniques.<br>e) Develop a Web2Py-based dashboard.<br>f) Store data in SQL/NoSQL database. | 10 |
| | **Total Hours** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262E | **ROBOTICS** | PECP | 2 | 0 | 2 | **3** | 2026 |

**i) COURSE OBJECTIVES:**

This course introduces the parts of robots, basic working concepts and types of robots. It makes the students familiar with machine operations using robots and discusses the applications and implementation of robot control systems.

**ii)COURSE OUTCOMES:**

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

| | | |
|---|---|---|
| CO1 | Explain the working principle of robots | Understand |
| CO2 | Identify the purpose of various sensors in robots for automation. | Apply |
| CO3 | Develop robotic arm to handle the materials and machines | Apply |
| CO4 | Develop robot programming for controlling robots. | Apply |
| CO5 | Demonstrate the application of robots in industry. | Apply |

**iii) SYLLABUS**

Introduction, Robot Kinematics, Actuators and Control, Programmable Logic Controllers, Servo control in a Robot, Applications of Robots

**iv) REFERENCES:**

1. John Craig, " Introduction to Robotics, Mechanics and Control" February 2017, Pearson

2. S.R. Deb, "Robotics technology and flexible automation", THH-2009.

3. Saeed B.Nikku, Introduction to robotics, analysis, control and applications, Wiley-India, 2nd edition 2011.

4. Richared D.Klafter. Thomas Achmielewski and Mickael Negin, Robotic Engineering and Integrated Approach, Prentice Hall India-New Delhi-2001.

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Introduction to Robotics, robot anatomy, degrees of freedom, types of robots, applications, safety, AI in robotics. | 5 |
| II | Robot kinematics, forward and inverse kinematics, robot arm configuration, mobile robots. | 5 |
| III | Actuators and sensors: DC motor, stepper motor, servo motor, IR, ultrasonic, proximity sensors and end effectors. | 8 |
| IV | Robot control: PID control, servo control, PLC basics, PC-based control systems | 5 |
| V | Industrial robotics, automation levels, material handling, inspection systems, real-world applications. | 7 |
| | **Total hours** | 30 |

**LAB / PRACTICAL**

| | Experiment | No. of hours |
|---|---|---|
| **Module-1** | | |
| | 1. Introduction to robotics platform. (Arduino kit)<br>2. LED and buzzer control using microcontroller. | **4** |
| **Module-2** | | |
| | 3. Control of DC motor using motor driver.<br>4. Servo motor position control for robotic joint. | **4** |
| **Module-3** | | |
| | 5. Interfacing IR sensor for obstacle detection.<br>6. Ultrasonic sensor for distance measurement. | **4** |
| **Module-4** | | |
| | 7. Line-following robot using IR sensors.<br>8. Obstacle-avoiding robot using ultrasonic sensor. | **4** |
| **Module-5** | | |
| | 9. Localization and Navigation of a Mobile Robot Using ROS. | **4** |
| **Project** | | |
| | **Project:** Design and implement a simple robotic system that includes:<br>• Sensor input (IR/ultrasonic/vision)<br>• Actuator output (motor/servo) | **10** |

| | | |
|---|---|---|
| | • Microcontroller-based control logic<br>• Demonstration of a real-world application | |
| | **Total hours** | 30 |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262K | **NATURAL LANGUAGE PROCESSING** | PECP | 2 | 0 | 2 | **3** | 2026 |

**i) COURSE OBJECTIVES**:

This course provides an in-depth and research-oriented study of Natural Language Processing, covering linguistic foundations, probabilistic and neural modeling techniques, and large language models. Students will critically analyze, design, and evaluate NLP systems while addressing ethical, multilingual, and real-world deployment challenges.

ii) **COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Apply linguistic representations and text preprocessing techniques to improve the performance of NLP models | Apply |
| CO2 | Develop statistical language models and classical machine learning methods for text classification and language modelling tasks. | Apply |
| CO3 | Apply probabilistic sequence models and information extraction techniques such as Named Entity Recognition and spelling correction | Apply |
| C04 | Apply syntactic and semantic structures using parsing, word sense disambiguation, and semantic role labeling to address language ambiguity. | Apply |
| CO5 | Build neural network–based NLP models using word embeddings, recurrent architectures, and transformer-based pretrained models for real-world applications. | Apply |

**iii) SYLLABUS**

**Fundamentals of NLP and Linguistic Concepts:** NLP definition and applications – challenges in NLP – linguistic levels: morphology – syntax – semantics – pragmatics – discourse – linguistic ambiguity – lexical semantics – lexical resources: WordNet – VerbNet.
**Text Processing and Representation:** Tokenization – stemming – lemmatization – stop-word removal – normalization – Part-of-Speech tagging – Bag-of-Words – TF-IDF.
**Statistical and Traditional Machine Learning Approaches:** Language modeling – n-gram models – smoothing techniques – perplexity – text classification – Naïve Bayes – Logistic Regression – Support Vector Machines – sequence labeling – Hidden Markov Models – Conditional Random Fields.
**Parsing and Semantic Processing:** Constituency parsing – dependency parsing – context-free

grammars – probabilistic parsing – CYK algorithm – word sense disambiguation – semantic role labeling – coreference resolution – discourse analysis – syntax–semantics interface.

**Deep Learning for NLP:** Word embeddings: Word2Vec – GloVe – FastText – neural networks for NLP – RNNs – LSTMs – GRUs – sequence-to-sequence models – attention mechanism – transformer architecture – contextual embeddings – pretrained language models: BERT – GPT – T5 – fine-tuning – transfer learning.

**LLMs,Applications, and Ethics:** Large language models – prompt engineering – multilingual NLP – bias – fairness – explainability – privacy – responsible and ethical deployment of NLP systems.

**iv) REFERENCES**


- Speech and language processing (3rd ed., draft). Jurafsky, D., & Martin, J. H. (2023). Pearson Education.
- Natural language processing with Python: Analyzing text with the natural language toolkit. Bird, S., Klein, E., & Loper, E. (2009). O’Reilly Media.
- Deep learning for natural language processing. Goyal, P., Pandey, S., & Jain, K. (2018). Apress.
- Natural Language Processing with Transformers. Tunstall, L., von Werra, L., & Wolf, T. (2022). O’Reilly Media.
- Fairness and Machine Learning: Limitations and Opportunities. Barocas, S., Hardt, M., & Narayanan, A. (2023). MIT Press.
- Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots. McTear, M. (2020). Morgan & Claypool Publishers.


**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Natural Language Processing: scope, importance, and research challenges. Applications in healthcare, social media, finance, and legal systems. Linguistic foundations: morphology, syntax, semantics, pragmatics, and discourse. Morphological analysis: stemming, lemmatization, and morphological parsing. Text preprocessing: tokenization, sentence segmentation, normalization, stop-word removal, handling noisy and code-mixed data. Part-of-Speech tagging. Corpus design and development: multilingual and domain-specific corpora. Annotation standards, annotation schemes, inter-annotator agreement, annotation bias, and linguistic ambiguity. | 6 |
| II | Statistical language modeling and N-gram models: unigram, bigram, trigram. Data sparsity problem and smoothing techniques: Laplace and Good–Turing estimation. Language model evaluation: perplexity. Text representation: Bag-of-Words, TF-IDF, and feature engineering. Classical text classification models: Naïve Bayes, Logistic Regression, Support Vector Machines. Evaluation metrics: precision, recall, F1-score. Sequence labeling tasks: Named Entity Recognition and POS tagging. Probabilistic sequence models: Hidden Markov Models and Conditional Random Fields. | 6 |

| | | |
|---|---|---|
| III | Syntactic parsing: dependency parsing, constituency parsing, context-free grammars, probabilistic parsing, CYK algorithm, parse trees. Semantic analysis: word sense disambiguation, semantic similarity, semantic role labeling. Lexical and semantic resources such as WordNet, VerbNet, and FrameNet. Discourse processing: coherence, structure, and coreference resolution. Knowledge representation: ontologies, first-order predicate logic, and knowledge graphs. Semantic search and question answering. | 6 |
| IV | Representation learning and distributed word representations. Word embeddings: Word2Vec, GloVe, FastText, contextual embeddings. Recurrent Neural Networks, Long Short-Term Memory, and Gated Recurrent Units. Sequence-to-sequence models, encoder–decoder architecture, and attention mechanisms. Applications in machine translation, summarization, and text generation. | 6 |
| V | Transformer architecture and self-attention. Pretrained language models such as BERT and RoBERTa. Transfer learning and fine-tuning. Multilingual and cross-lingual NLP, low-resource languages, and code-mixed data. Cross-lingual transfer models such as mBERT and XLM-R. Ethical and responsible AI: bias, fairness, explainability, and privacy. Emerging applications: conversational agents, social media analysis, sentiment and hate speech detection. | 6 |
| | **Total hours** | **30** |

**LAB / PRACTICAL**

| | Experiment | No. of hours |
|---|---|---|
| **Module-1** | | |
| 1 | 1) Perform text preprocessing on a given dataset. Implement tokenization, normalization, stop-word removal, stemming, and lemmatization.<br>2) Create a small text corpus and perform annotation for tasks such as Part-of-Speech tagging or sentiment labeling. Evaluate inter-annotator agreement. | 6 |
| **Module2** | | |
| 2 | 1) Implement N-gram language models and apply smoothing techniques. Evaluate the model using perplexity.<br>2) Develop a text classification system using classical machine learning algorithms such as Naïve Bayes or Logistic Regression. Evaluate the model using precision, recall, and F1-score. | 6 |
| **Module3** | | |

| | | |
|---|---|---|
| 3 | 1) Perform syntactic parsing and generate dependency or constituency parse trees.<br>2) Implement Word Sense Disambiguation or semantic similarity using lexical resources such as WordNet. | 6 |
| **Module-4** | | |
| 4 | 1) Train word embeddings such as Word2Vec or FastText and evaluate their performance.<br>2) Develop a deep learning model such as RNN or LSTM for text classification or sentiment analysis. | 6 |
| **Module-5-Project** | | |
| 5 | 1) Develop an NLP application, such as machine translation, chatbot, or text summarization, etc., by fine-tuning pretrained models such as BER. | 6 |
| | **TOTAL HOURS** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262C | **TRUSTWORTHY ARTIFICIAL INTELLIGENCE** | PEC | 3 | 0 | 0 | **3** | 2026 |

**i) COURSE OBJECTIVES**

To apply principles, techniques, and frameworks of Trustworthy Artificial Intelligence, Responsible AI, and Federated Learning for designing secure, fair, privacy-preserving, and robust AI systems.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Apply Responsible AI principles to design AI systems that ensure fairness, transparency, ethics, and accountability | Apply |
| CO2 | Apply threat modeling and risk analysis techniques to identify reliability, safety, and robustness issues in AI systems | Apply |
| CO3 | Apply robustness and defense mechanisms to machine learning, deep learning, and large language models | Apply |
| CO4 | Apply privacy-preserving learning techniques, including Federated Learning and Differential Privacy, in centralized and decentralized AI systems | Apply |
| CO5 | Apply provenance, explainability, evaluation, and regulatory frameworks to assess the trustworthiness of generative AI systems | Apply |

**iii) SYLLABUS**

Foundations of Trustworthy and Responsible Artificial Intelligence, reliability, safety, security, privacy, and robustness in AI systems, ethical and responsible AI principles, threat models and failure modes in machine learning, adversarial attacks and defenses in deep learning and large language models, certified robustness and verification techniques, robustness challenges in generative and agentic AI systems, privacy threats including data leakage, model inversion, poisoning, and membership inference, privacy-preserving machine learning, federated learning architectures and secure aggregation, differential privacy and its applications, memorization and training data extraction in generative AI models, private attribute inference, provenance and detection of AI-generated content, watermarking techniques and attacks on watermarking, dataset contamination and integrity, trustworthy evaluation and benchmarking of large language models, and alignment of technical evaluation with AI governance and regulatory frameworks.

### iv) REFERENCES


1. Dan Hendrycks, *Introduction to AI Safety, Ethics, and Society*, Chapman & Hall / CRC Press, 2025
2. Beena Ammanath, *Trustworthy AI: A Business Guide for Navigating Trust and Ethics in AI,* Wiley, 2022
3. Luciano Floridi, *The Ethics of Artificial Intelligence: Principles, Challenges, and Opportunities*, Oxford University Press, 2023
4. Barocas, S., Hardt, M., and Narayanan, A., *Fairness and Machine Learning: Limitations and Opportunities,* MIT Press, 2023.
5. A. Joseph, B. Nelson, B. Rubinstein, D. Tygar, *Adversarial machine learning*, Cambridge University Press, 2019
6. Kush R. Varshney, *Trustworthy Machine Learning*, Independently Published, 2022


### v) COURSE PLAN

| Module | Contents | No. of hours |
|---|---|---|
| I | **Foundations of Trustworthy and Responsible AI**<br>Introduction to Trustworthy and Responsible Artificial Intelligence. Ethical, legal, and societal motivations for Responsible AI. Reliability, safety, security, privacy, and robustness as core challenges in AI systems. Trust and accountability in AI decision-making. Threat models in machine learning. Risk analysis and failure modes of AI systems. Overview of adversarial learning and privacy-preserving learning. Trustworthiness requirements across the AI lifecycle. Case studies illustrating trust failures and ethical concerns in deployed AI systems. | 9 |
| II | **Robustness in Machine Learning and Deep Learning**<br>Adversarial attacks on machine learning and deep learning models including evasion attacks, poisoning attacks, and backdoor attacks. Defense mechanisms against adversarial attacks. Certified robustness of neural networks. Automated certification techniques such as convex relaxations, interval bound propagation, branch and bound methods, and randomized smoothing. Certified training of deep neural networks using symbolic and continuous optimization methods. Robustness challenges in large-scale, foundation, and safety-critical AI systems. | 9 |
| III | **Trustworthy Large Language Models and Generative AI**<br>Trust and robustness issues in generative AI systems. State-of-the-art attacks and novel attack vectors for large language models including prompt injection, jailbreak attacks, hallucinations, data leakage, and model manipulation. Robust prompting and alignment strategies for LLMs. Safety challenges in agentic and tool-using AI systems. Securing data flows and interactions in agent-based AI. Case studies on robustness and safety failures in generative AI deployments. | 9 |

| | | |
|---|---|---|
| IV | **Privacy-Preserving and Federated Learning**<br>Privacy threat models including data stealing, model inversion, model poisoning, membership inference, and property inference attacks. Privacy attacks in centralized and decentralized machine learning. Privacy challenges in collaborative and federated learning environments. Federated learning architectures, secure aggregation, and communication protocols. Integration of federated learning with Responsible and Trustworthy AI frameworks. Differential privacy fundamentals, privacy budgets, and guarantees. Differentially private training for centralized and federated models. Memorization and training data extraction attacks in generative AI. Private attribute inference using AI models. | 9 |
| V | **Provenance, Evaluation, and Regulation of AI Systems**<br>Provenance in AI systems and its role in trust and accountability. Detection of AI-generated content. Watermarking techniques for generative models. Robust and fragile watermarking for text, image, and multimedia data. Attacks on watermarking including removal, forgery, and evasion. Dataset contamination and poisoning in AI pipelines. Detection of contaminated datasets and evasion techniques. Trustworthy evaluation and benchmarking of large language models. Challenges in rating and comparing generative AI systems. Bridging AI regulation<br><br>(e.g., EU AI Act) with technical evaluation, auditing, and governance mechanisms. | 9 |
| | **Total hours** | **45** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS162D | **DEEP LEARNING TECHNIQUES** | PECP | 2 | 0 | 2 | **3** | 2026 |

## i) COURSE OBJECTIVES

This course introduces the core concepts of deep learning and also provides an insight into recent developments in the field. The concepts covered in the course include Neural Network Optimization techniques, Regularization, Convolutional Neural networks, Recurrent Neural Networks, Transformers and Object Detection. This course helps the students to develop solutions to real world applications using deep learning techniques.

## ii) COURSE OUTCOMES

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

| | | |
|---|---|---|
| CO1 | Apply the standard regularization and optimization techniques for the effective training of deep neural networks. | Apply |
| CO2 | Illustrate the working of probabilistic and generative DL models like Variational Autoencoders, Generative Adversarial Networks. | Apply |
| CO3 | Make use of recurrent neural networks and its variants in relevant application areas. | Apply |
| CO4 | Apply the transformer architecture and compare it with earlier architectures. | Apply |
| CO5 | Identify solutions to real world problems by applying deep learning techniques. | Apply |

## iii) SYLLABUS

Neural network training methods, optimization techniques, and regularization. Convolutional Neural Networks (CNNs) and popular architectures for image classification. Recurrent Neural Networks (RNNs), including LSTM and GRU, for sequential data modeling. Transformer-based models, including attention, object detection and segmentation.

## iv) TEXTBOOKS

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016
2. M.Gopal, Deep Learning, Pearson , 2022
3. Aston Zhang, Zachary C.Lipton, Mu Li ,and Alexander J.Smola, Dive into Deep Learning, available online at **https://d2l.ai**.

.

**REFERENCES**

1. Bishop, C,M , Pattern Recognition and Machine Learning, Springer, 2006
2. Russell Reed and Robert J. Marks II, Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks, Bradford Book, 2014.
3. Michael Nielsen, Neural Networks and Deep Learning, 2015

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Deep Learning vs traditional machine learning, Gradient Descent, Adam Optimization, Weight initialization strategies, Batch Normalization. Convolutional Neural Networks: – convolution operation, pooling,Relation between input size, output size and filter size, 3D Convolution, Backpropagation in Convolutional Layers. | 7 |
| II | Convolutional Architectures: GoogleNet, VGG, ResNet, EfficientNet, Transfer Learning. Deep Autoencoders- sparse autoencoders, Denoising autoencoders. Introduction to GAN , Vanilla GAN, Diffusion Models. | 6 |
| III | Recurrent Neural Networks, Back propagation through time , Vanishing and exploding gradients with RNNs, Different types of RNNs-overview, Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU). Applications of RNN: word prediction, Chatbots, Image captioning . | 6 |
| IV | Attention Model - Multi-head attention, Self-attention and positional encoding, Transformer architecture, Transformers for vision, Large-scale pretraining with transformers | 6 |
| V | Instance Segmentation v/s Semantic Segmentation, Object Detection : RCNN , Fast RCNN, Faster RCNN , YOLO, Mask RCNN | 5 |
| | **Total hours** | **30** |

| Exp No. | Contents | No. of hours |
|---|---|---|
| 1. | Analyze the impact of optimization, weight initialization techniques, dropout and regularization techniques, and visualize the change in performance. | 3 |
| 2. | Implement a CNN with at least 2 convolutional layers to classify images into binary classes (e.g., even vs odd digits for MNIST, or T-shirt vs trouser | 3 |

| | | |
|---|---|---|
| | for Fashion MNIST). Include padding and Batch Normalization, and record accuracy vs epochs. | |
| 3. | Design and implement a CNN model (with 4+ layers of convolutions) to classify multi category image datasets. Use the concept of regularization and dropout while designing the CNN model. Use the Fashion MNIST datasets. Record the Training accuracy and Test accuracy corresponding to the following architectures:<br>a. Base Model<br>b. Model with L1 Regularization<br>c. Model with L2 Regularization<br>d. Model with Dropout<br>e. Model with both L2 (or L1) and Dropout | 3 |
| 4. | Digit classification using pre-trained networks like VGGnet-19 or ResNet for MNIST dataset and analyze and visualize performance improvement. | 3 |
| 5. | Implement a shallow autoencoder for image reconstruction using MNIST dataset. | 3 |
| 6. | Implement image generation using GAN. | 3 |
| 7. | Implement a simple RNN. Analyze and visualize the performance change while using LSTM and GRU instead of simple RNN. | 3 |
| 8. | Implement a Transformer-based classifier in PyTorch for the MNIST (or Fashion-MNIST) dataset and train it to classify images into 10 classes. | 3 |
| 9. | Object detection using YOLO and Faster RCNN. | 3 |
| 10. | Implement image segmentation using Mask R-CNN | 3 |
| | **Total hours** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262H | **GENERATIVE AI** | PECP | 2 | 0 | 2 | **3** | 2026 |

**i) COURSE OBJECTIVES**

The Generative AI (GenAI) syllabus presents the core principles, models, and techniques that enable AI systems to generate original content such as text, images, audio, and code. It builds on foundational concepts from classical AI and machine learning to highlight the shift from prediction-oriented models to generative approaches. The syllabus examines prominent language models, including GPT, BERT, and T5, focusing on their training mechanisms and practical applications. It also covers Retrieval-Augmented Generation (RAG) for enhancing model outputs with external knowledge, along with prompt engineering methods for effective interaction with generative models. Overall, the course aims to develop students' ability to design, evaluate, and apply GenAI systems in a responsible and innovative manner.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Implement appropriate machine learning, deep learning, and generative AI models based on data type and problem context. | Apply |
| CO2 | Utilize foundational language models and retrieval-augmented generation techniques for domain-specific tasks. | Apply |
| CO3 | Employ transformer-based models and tokenization techniques to generate text and speech outputs. | Apply |
| CO4 | Apply diffusion-based generative models for image and video generation in real-world applications. | Apply |
| CO5 | Apply agentic AI frameworks, vector databases, and advanced prompting strategies to build generative AI applications. | Apply |

**iii) SYLLABUS**

Overview of Classical Machine Learning and Artificial Intelligence, Difference between Gen AI and other types of AI, Advantages and disadvantages of Gen AI technologies, Foundational Language Models, Retrieval Augmented Generation RAG Framework, Generative Text and Speed Models, Generative Image models, Prompt Engineering, and Agentic AI.

**iv) TEXT BOOKS**

1. Modern Generative AI with ChatGPT and OpenAI Models: Leverage the Capabilities of OpenAI's LLM for Productivity and Innovation with GPT3 and GPT4, by Valentina Alto, Packt Publishing Ltd, 2023.

2. Generative AI for Cloud Solutions: Architect modern AI LLMs in secure, scalable, and ethical cloud environments, by Paul Singh, Anurag Karuparti ,Packt Publishing Ltd, 2024.

**REFERENCES**

1. Foster, D. (2023). Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play (2nd ed.). O'Reilly Media. ISBN: 978-1-098-13418-1.
2. Tunstall, L., von Werra, L., & Wolf, T. (2022). Natural Language Processing with Transformers: Building Language Applications with Hugging Face. O'Reilly Media. ISBN: 978-1-098-10324-8

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Overview of Classical Machine Learning and Artificial Intelligence<br>Data Types and State of the Art models<br>• Tabular Data - Gradient Boosted Models<br>• Image Data - Convolutional Neural Networks<br>• Sequential and Time Series Data - Recurrent Neural Networks<br>• Text and Speech Data – Transformers<br>Generative AI- GPT class of Models for Text, Diffusion for Images/Video, Difference between Gen AI and other types of AI, Advantages and disadvantages of Gen AI technologies. | 6 |
| II | Foundational Language Models LLAMA3 Instruct 8B/70B, LLAMA3 Chat, LLAMA3 code, E5 embedding, MIXTRAL 8X7B, SLMs (PHI3, bitNet B1.58).<br>LLM Fine tuning - Para Efficient Fine tuning PEFT (LORA, P tuning, Finetuning of embedding models)<br>Retrieval Augmented Generation<br>Alignment- RLHF, DPO, RPO. RAGs -Advanced ingestion, Chunking, Embedding, Search, Ranking, Generation, Evaluation.<br>RAG frameworks - LLANGCHAIN basics, LLAMAINDEX, LLANGRAPH. | 6 |
| III | Generative Text and Speech Models - Tokenization Fundamentals and Byte Pair Encoding, GPT class of models to Generate Text, Training GPT Models, Speech Models, Interacting with Trained Models. | 6 |
| IV | Generative Image models – Stable Diffusion Fundamentals, Image and Video Generation, Tools for Generating Images, GenAI Use cases. | 6 |
| V | Agentic AI, Retrieval Augmented Generation, and Advanced Prompt Engineering, Vector Database, LangChain with RAG and LLM Agents, | 6 |

| | | |
|---|---|---|
| | Advanced Prompting Strategies (e.g., CoT, ReAct, DSP), Basic Prompting to Build AI Applications, Hosting GenAI. | |
| | **Total** | **30** |

| Module | Exp No. | Contents | No. of hours |
|---|---|---|---|
| I | 1 | Implement Gradient Boosting for structured data classification. (Datasets: Breast Cancer / UCI Tabular dataset) | 2 |
| I | 2 | Build a CNN for image data classification. (Datasets: MNIST, Fashion-MNIST (toy), CIFAR-10, Chest X-ray (NIH) or RSNA Pneumonia for medical imaging.) | 2 |
| I | 3 | Apply RNN for time-series prediction. (Datasets: Synthetic sine waves for pedagogy, UCI Electricity load or M4 time-series subset for forecasting.) | 2 |
| II | 4 | Load and generate text using a foundation LLM - Llama3 / Mixtral. (Datasets: Small instruction tuning sets (Self-Instruct samples), domain QA pairs (medical textbook excerpts). Use Hugging Face model repos to load weights.) | 2 |
| II | 5 | Generate sentence embeddings using E5. (Dataset: MS-MARCO / Wikipedia passage subsets / custom course notes. Use E5 embedding models (intfloat) via Hugging Face or sentence-transformers.) | 2 |
| II | 6 | Implement basic Retrieval-Augmented Generation pipeline (LangChain + Vector DB). (Dataset: Small knowledge base built from course PDF slides, PubMed abstracts (subset), or local lab manuals.) | 2 |
| III | 7 | Implement Tokenization and BPE Analysis. ( Use any short corpora: Wikipedia paragraphs, medical abstracts, or code snippets to show tokenizer differences.) | 2 |
| III | 8 | Generate text using GPT-style models with prompts for summarization: research abstracts, lecture notes, healthcare case summaries. Evaluate coherence, factuality. (Use objective metrics (ROUGE/BLEU) + human evaluation) | 2 |
| III | 9 | Convert speech to text.using Whisper. (Dataset: Librispeech subsets, short recorded lectures, clinical speech) | 2 |
| IV | 10 | Generate images from text prompts using Stable Diffusion. | 2 |

| | | | |
|---|---|---|---|
| | | (Example of domain text: medical illustrations, microscopy styles etc.) | |
| **IV** | **11** | Apply image to image generation by modifying images using diffusion.<br><br>(Paired/unaligned datasets (e.g., CycleGAN datasets, histopathology patches). | 2 |
| **IV** | **12** | Analyze GenAI applications in healthcare using medical image synthesis.<br><br>(Public medical image datasets (ISIC for skin lesions, BraTS for brain tumor MRI)) | 2 |
| **V** | **13** | Store and retrieve embeddings using Vector DB (FAISS / Milvus / Pinecone).<br><br>(Use Wikipedia section dumps, course slide corpus, or PubMed subsets for domain retrieval.) | 2 |
| **V** | **14** | Build LLM agents using ReAct Tool wrappers (calculator, web search mock, local KB), prompt templates and Test the agent on multi-step tasks. | 2 |
| **V** | **15** | Compare Zero-shot, CoT, and ReAct prompts.<br><br>(Use the benchmarks with reasoning datasets (GSM8K for math reasoning), domain Q&A sets (course FAQs)) | 2 |
| | | **Total** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262I | **AGENTIC AI** | PEC | 3 | 0 | 0 | **3** | 2026 |

**i) COURSE OBJECTIVES**

This course aims to provide an in-depth understanding of agentic artificial intelligence systems that can autonomously perceive, reason, plan, and act in dynamic environments. It focuses on the design and analysis of intelligent agent architectures, learning mechanisms, and multi-agent interactions. The course also prepares students to develop and evaluate modern agentic AI applications while addressing ethical, safety, and real-world deployment challenges.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Explain the fundamental principles of Agentic AI, intelligent agents, and agent–environment interactions. | Understand |
| CO2 | Describe intelligent agent architectures using appropriate reasoning and knowledge representation techniques. | Understand |
| CO3 | Explain the principles of planning, learning, and decision-making agents. | Understand |
| CO4 | Apply principles of multi-agent systems to model and solve distributed problems by using agent communication languages (ACL, KQML, FIPA). | Apply |
| CO5 | Apply concepts of LLM-based and agentic AI systems to build and evaluate tools. | Apply |

**iii)SYLLABUS**

Introduction to Agentic AI. Intelligent Agents: Definition, Characteristics, and Types. Agent–Environment Interaction Models. Rationality, Autonomy, Reactivity, Proactiveness. PEAS Framework and Environment Types. Comparison: Agentic AI vs Traditional AI vs Generative AI. Applications of Agentic AI in Industry and Research.

Agent Architectures-Simple Reflex Agents- Model-Based Agents- Goal-Based Agents- Utility-Based Agents-Learning Agents
Knowledge Representation for Agents. Logic-Based Reasoning and Inference. BDI (Belief–Desire–Intention) Architecture. Reactive vs Deliberative Agents. Hybrid Agent Architectures

Automated Planning in Agentic Systems. State-Space Search and Planning Algorithms. Markov Decision Processes (MDP). Partially Observable MDPs (POMDPs). Reinforcement Learning for Agents. Hierarchical Reinforcement Learning. Goal-Conditioned and Skill-Based Agents. Reward Design and Policy Optimization

Introduction to Multi-Agent Systems. Agent Communication Languages (ACL, KQML, FIPA). Cooperation, Coordination, and Negotiation. Distributed Problem Solving. Game Theory for

Multi-Agent Systems. Competitive vs Cooperative Agents. Swarm Intelligence (Ant, PSO, Flocking Models). Trust, Ethics, and Safety in MAS.

Large Language Model (LLM) based Agents. Tool-Using and Autonomous Agents (AutoGPT, BabyAGI concepts). Memory, Planning, and Reflection in Agents. Human-in-the-Loop Agentic Systems. Agentic AI Frameworks (LangChain, CrewAI, AutoGen – conceptual). Evaluation Metrics for Agentic Systems. Case Studies: Robotics, Autonomous Vehicles, Smart Systems. Future Directions and Research Challenges in Agentic AI.

**iv) TEXT BOOKS**


1. Russell, Stuart J and Norvig, Peter. Artificial Intelligence: A Modern Approach. 4th Edition, Pearson Education Limited, Harlow, England, 2020.

2. Wooldridge, Michael. An Introduction to MultiAgent Systems. 2nd Edition, John Wiley & Sons, Chichester, United Kingdom, 2009.


**v) REFERENCES**


1. https://arxiv.org/abs/2601.12560
2. https://arxiv.org/abs/2509.06283


**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | **Foundations of Agentic AI**<br>Introduction to Agentic AI. Intelligent Agents: Definition, Characteristics, and Types. Agent–Environment Interaction Models. Rationality, Autonomy, Reactivity, Proactiveness. PEAS Framework and Environment Types. Comparison: Agentic AI vs Traditional AI vs Generative AI. Applications of Agentic AI in Industry and Research | 9 |
| II | Agent Architectures and Reasoning<br>Agent Architectures-Simple Reflex Agents- Model-Based Agents-Goal-Based Agents- Utility-Based Agents-Learning Agents<br>Knowledge Representation for Agents. Logic-Based Reasoning and Inference. BDI (Belief–Desire–Intention) Architecture. Reactive vs Deliberative Agents. Hybrid Agent Architectures | 9 |
| III | Planning, Learning, and Decision-Making Agents<br>Automated Planning in Agentic Systems. State-Space Search and Planning Algorithms. Markov Decision Processes (MDP). Partially Observable MDPs (POMDPs). Reinforcement Learning for Agents. Hierarchical Reinforcement Learning. Goal-Conditioned and Skill-Based Agents. Reward Design and Policy Optimization | **9** |
| IV | Multi-Agent Systems (MAS)<br>Introduction to Multi-Agent Systems. Agent Communication Languages (ACL, KQML, FIPA). Cooperation, Coordination, and Negotiation. Distributed Problem Solving. Game Theory for Multi-Agent Systems. Competitive vs Cooperative Agents. Swarm Intelligence (Ant, PSO, Flocking Models). Trust, Ethics, and Safety in MAS. | **9** |
| V | **Modern Agentic AI Systems and Applications** | **9** |

| | | |
|---|---|---|
| | Large Language Model (LLM) based Agents. Tool-Using and Autonomous Agents (AutoGPT, BabyAGI concepts). Memory, Planning, and Reflection in Agents. Human-in-the-Loop Agentic Systems. Agentic AI Frameworks (LangChain, CrewAI, AutoGen – conceptual). Evaluation Metrics for Agentic Systems. Case Studies: Robotics, Autonomous Vehicles, Smart Systems. Future Directions and Research Challenges in Agentic AI. | |
| | **Total** | **45** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS262J | **AI DRIVEN THREAT ANALYSIS FOR CYBER SECURITY** | PEC | 2 | 0 | 2 | **3** | 2026 |

**ii) COURSE OBJECTIVES**

The purpose of this course is to provide the learner a comprehensive understanding of how artificial intelligence (AI) is transforming the field of threat analysis in the context of cyber security. The course explores cutting-edge applications of AI in anomaly analysis, detection and providing predictive analytics. Necessary tools to analyze threats, protect valuable data, malware analysis, cloud security and incident response will also be covered in the course.

**iii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Explain the fundamental concepts, principles, and technologies used in AI-driven threat analysis for cybersecurity. | Understand |
| CO2 | Explain various types of threats, attack vectors, and vulnerabilities commonly exploited by malicious actors. | Understand |
| CO3 | Apply AI techniques to detect, analyze, and mitigate cyber threats, enabling proactive defense and timely incident response. | Apply |
| CO4 | Develop skills in gathering, analyzing, and interpreting threat intelligence to proactively identify and track potential threats. | Apply |
| CO5 | Apply AI-driven threat analysis techniques in cyber security | Apply |

**iv) SYLLABUS**

**Introduction to cybersecurity and threat intelligence.** Role of AI in cybersecurity, AI trust and explainability, cybersecurity life cycle, and learning technologies. Anomaly and multi-attack detection using cyber learning. Basics of modern malware detection and analysis.

**Identification of assets, security threats, and vulnerabilities.** Secure coding practices and vulnerability analysis. Overview of OWASP Top 10 security risks, including access control, cryptographic failures, injection, insecure design, misconfiguration, outdated components, authentication failures, and integrity issues.

**Cryptography and hashing techniques used in industry.** AI-based threat detection including malware, network anomalies, and botnet detection using machine learning. Protection of sensitive information, secure authentication mechanisms, and evaluation of AI models for data quality, security, and reliability.

**Security assessment and penetration testing concepts**, PTES phases, types of penetration tests, tools, and vulnerability scanners. Cloud security fundamentals including SPI framework, infrastructure security, data security, identity and access management, and cloud security management.

**Cybersecurity ontologies and knowledge modeling**. Incident response and digital forensics techniques including memory forensics, log analysis, and malware analysis. Hands-on incident response investigation. Emerging trends in cybersecurity such as AI, IoT, and blockchain security

**v) REFERENCES**


1. AI-Driven Cybersecurity and Threat Intelligence, Sarker, Iqbal H. Springer Nature Switzerland, 2024.
2. Cloud security and privacy, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly Media, Inc., 2009
3. Hands-On Artificial Intelligence for Cybersecurity, Alessandro Parisi, Packt Publishing, 2019.
4. "AI in Cybersecurity", Leslie F. Sikos(Editor), Springer International Publishing, 2018
5. “Security Software Development Assessing and Managing Security Risks”, Douglas A. Ashbaugh, CRC Press, 2008
6. “Metasploit: The Penetration Tester's Guide”, David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni, No Starch Press,US, 2011
7. “Incident Response and Computer Forensics”, Kevin Mandia, Mathew Pepe, Jason Luttgens,3rd Edition, McGraw-Hill Osborne Media, 2014.
8. Emerging ICT Technologies and Cybersecurity, Thakur, Kutub, Al-Sakib Khan Pathan, and Sadia Ismat. Cham, Switzerland: Springer, 2023.


**vi) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | Cybersecurity and Threat Intelligence- Understanding Artificial Intelligence (AI) in Cybersecurity- AI Trust, Explainability, and Key Factors- Cybersecurity Life Cycle- Various Types of Learning Technologies. Detecting Anomalies and Multi-attacks Through CyberLearning. | 5 |
| II | Identifying Assets- Identifying Security threats- Identifying Vulnerabilities-Secure coding techniques to avoid vulnerabilities-Vulnerabilities associated with the test case- OWASP-Broken Access Control Cryptographic failures- Injection- Insecure Design- Security Misconfiguration- Vulnerable and outdated components- Identification and Authentication failures. | 5 |
| III | Cryptography & Hashing- Different Cryptographic methods used in Industry- Hashing methods. Detecting Cybersecurity Threats with AI-Malware Threat Detection- -Different ML algorithm for botnet detection. Protecting Sensitive Information and Assets- Securing User Authentication. | 7 |
| IV | Security Assessment and Penetration Testing- Phases of the PTES-Types of Penetration Tests- Exploring tools and techniques for security assessment and penetration testing -Vulnerability Scanners. Cloud | 7 |

| | | |
|---|---|---|
| | Security- SPI Framework for Cloud Computing- Infrastructure Security- Data Security And Storage | |
| V | OWL Ontologies in Cybersecurity: Conceptual Modeling of Cyber Knowledge- Domain Ontologies for Cybersecurity, Incident Response and Forensics- Advanced incident response techniques and forensic investigation methodologies- Understanding memory forensics, log analysis, and malware analysis in incident response | 6 |
| | **Total hours** | **30** |

**LAB / PRACTICAL**

| | Experiment | No. of hours |
|---|---|---|
| **MODULE - 1** | | |
| **1** | Network Discovery and Mapping<br>● Utilize tools like Nmap and Wireshark to perform network discovery.<br>● Create a visual map of the network infrastructure.<br>● Analyze the implications of the network structure on management strategies. | **4** |
| **MODULE - 2** | | |
| **2** | Installation of Cyber Security Libraries using pip and execution of a basic Python security script<br>● Password Hashing and Network Scanning<br>● Simple Website Status Checker (Using requests)<br>● Demonstrate detection of common OWASP vulnerabilities such as SQL Injection, XSS, and weak passwords. | **6** |
| **MODULE -3** | | |
| **3** | ● Data Handling, Analysis, and Predictive Modeling using Python<br>● Data Loading, NumPy Operations, and Pandas Data Analysis<br>● Spam Mail detection using Naïve Bayesian classifier | **6** |
| **MODULE -4** | | |
| **4** | Packet Sniffing | **4** |

| | | |
|---|---|---|
| | • Capture network packets using a library like Scapy. | |
| **MODULE – 5 PROJECT** | | |
| **5** | Conducting an advanced incident response and forensic investigation.<br>Exploring topics such as AI in cybersecurity, IoT security, and blockchain security. | **10** |
| | **Total Hours** | **30** |

# INDUSTRY ELECTIVE COURSE (IEC)

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS266A | **CLOUD COMPUTING FOR ARTIFICIAL INTELLIGENCE** | IEC | 2 | 0 | 2 | **3** | 2026 |

**i) COURSE OBJECTIVES**

This course provides a comprehensive introduction to machine learning (ML) techniques with a focus on cloud computing platforms. Students will learn how to leverage cloud services to build, train, and deploy machine learning models efficiently. The course covers various machine learning algorithms, cloud-based tools, and ML workflows from data preprocessing to deployment. Hands-on labs will enable students to work with popular cloud ML services such as Amazon SageMaker, Google AI Platform, and Azure Machine Learning, applying these tools to solve real-world problems at scale.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| **CO1** | Explain the principles of machine learning (ML) and how it integrates with cloud computing platforms. | Understand |
| **CO2** | Utilize cloud infrastructure for machine learning workflows, including data preprocessing, feature engineering | Apply |
| **CO3** | Utilize cloud-native machine learning tools and frameworks for developing and training models.. | Apply |
| **CO4** | Make use of cloud based services to deploy and scale machine learning models | Apply |
| **CO5** | Develop practical skills in applying machine learning to real-world applications in cloud environments, such as AWS, Google Cloud, and Azure. | Apply |

**iii) SYLLABUS**

Introduction to Machine Learning and Cloud Computing. Overview of machine learning, Machine learning in cloud environments, Overview of cloud platforms, Cloud-native machine learning services, Introduction to cloud-based data storage and computing resources for ML.Data Preprocessing and Feature Engineering in the Cloud-Data collection and storage in cloud environments, Data preprocessing techniques, Feature selection and extraction for machine learning models, Tools for data preprocessing in the cloud, Hands-on lab. Machine Learning Algorithms and Model Training in the Cloud-Overview of supervised, unsupervised, and reinforcement learning algorithms, Training machine learning models in the cloud, Distributed training, Hyperparameter tuning and optimization using cloud-based services, Hands-on lab: Training a machine learning model using Amazon SageMaker. Machine Learning Model Deployment and Monitoring in the Cloud-Deploying machine learning models in the cloud: Real-time and batch inference, Model serving and endpoint management in cloud environment  Monitoring model performance, Autoscaling machine learning models in cloud environments for real-time predictions. Hands-on lab. Advanced Topics in Machine Learning for Cloud Computing- Using AutoML for building machine learning models in the cloud., Machine learning pipelines, Security and privacy concerns in cloud-based machine learning ,Case studies, Future trends.

**iv) REFERENCES**


1. "Hands-On Machine Learning on Google Cloud Platform" by Giuseppe Ciaburro, Packt Publishing.
2. "Practical Deep Learning for Cloud, Mobile, and Edge: Real-World AI & Computer-Vision Projects Using Python, Keras & TensorFlow" by Anirudh Koul, O'Reilly Media.
3. "Data Science on the Google Cloud Platform" by Valliappa Lakshmanan, O'Reilly Media.
4. "Machine Learning Engineering in Action" by Ben Wilson, Manning Publications.
5. "Mastering Machine Learning on AWS" by Saket S. R. Mengle, Packt Publishing.


**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I<br>I | Introduction to Machine Learning and Cloud Computing,Overview of machine learning: Concepts, algorithms, and use cases. Machine learning in cloud environments: Benefits, challenges, and opportunities. Overview of cloud platforms: AWS, Azure, and Google Cloud. Cloud-native machine learning services: Amazon SageMaker, Google AI Platform, Azure Machine Learning. Introduction to cloud-based data storage and computing resources for ML. | 6 |
| II | Data Preprocessing and Feature Engineering in the Cloud-Data collection and storage in cloud environments: S3, Google cloud Storage, Azure Blob Storage. Data preprocessing techniques: Cleaning, normalization, encoding, and transformation. Feature selection and extraction for machine learning models. Tools for data preprocessing in the cloud: AWS Glue, Google Dataflow, Azure Data Factory. | 6 |

| | | |
|---|---|---|
| III | Machine Learning Algorithms and Model Training in the Cloud Overview of supervised, unsupervised, and reinforcement learning algorithms. Training machine learning models in the cloud: AWS SageMaker, Google AI Platform, Azure ML. Distributed training: Leveraging cloud infrastructure for parallel model training. Hyperparameter tuning and optimization using cloud-based services. | 6 |
| IV | Machine Learning Model Deployment and Monitoring in the Cloud Deploying machine learning models in the cloud: Real-time and batch inference. Model serving and endpoint management in cloud environments. Monitoring model performance: Model drift, retraining, and continuous monitoring.Autoscaling machine learning models in cloud environments for real-time predictions | 6 |
| V | Advanced Topics in Machine Learning for Cloud Computing. Using AutoML for building machine learning models in the cloud. Machine learning pipelines: Automating end-to-end workflows. Security and privacy concerns in cloud-based machine learning. Case studies: Real-world applications of machine learning in cloud computing(e.g., healthcare, finance, IoT). Future trends: Federated learning, edge computing, and AI/ML integration in cloud environments. | 6 |
| | **Total Hours** | **30** |

**LAB / PRACTICAL**

| Exp No: | Contents | No. of hours |
|---|---|---|
| 1. | Create a collaborative learning environment for a specific topic via Google Apps, utilizing Google Drive for e-books and key articles, Google Docs for collaborative editing, and Google Slides for presentations. | 4 |
| 2. | Install VirtualBox, set up a virtual machine with Windows or Linux, and analyse its configuration settings. | 3 |

| | | |
|---|---|---|
| 3. | Register with Amazon AWS, launch an EC2 instance using the Amazon Linux 2023 (AL2023) AMI, and provision an S3 bucket. Upload a "hello_world" file to the S3 bucket, then connect via SSH to the EC2 instance and list the bucket's objects. | 3 |
| 4. | Implement a case study on Amazon Elastic Cloud Services | 4 |
| 5. | Implement a  case study on Azure | 3 |
| 6. | Installation and Configuration of Just cloud. | 3 |
| 7. | Installation and Configuration of Hadoop/Eucalyptus | 3 |
| 8. | Executing Hadoop MapReduce Jobs | 4 |
| 9. | Exploring Manjrasoft Aneka Platform | 3 |
| | **Total hours** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS266B | **FOUNDATIONS AND CONSTRUCTION OF LARGE LANGUAGE MODELS** | IEC | 2 | 0 | 2 | **3** | 2026 |

Pre-requisite: Linear Algebra, Probability & Statistics, Python Programming and Machine Learning basics

**i) COURSE OBJECTIVES**

- To develop strong mathematical foundations required for building language models.
- To design and implement neural language models from first principles.
- To construct and train a Transformer-based language model from scratch.
- To evaluate, optimize and analyze the model performance.

**ii) COURSE OUTCOMES**

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

| | | |
|---|---|---|
| CO1 | Explain probabilistic foundations of language modeling. | Understand |
| CO2 | Develop Neural Network-based language models from scratch. | Apply |
| CO3 | Construct a Transformer architecture without using pretrained libraries. | Apply |
| CO4 | Make use of small scale language models to train the data | Apply |
| CO5 | Identify the scalability, computational cost and ethical risks of LLMs. | Apply |

**iii)SYLLABUS**

Probability theory for language modeling, n-gram models, maximum likelihood estimation, smoothing techniques, neural networks for NLP, embedding layers, backpropagation, optimization techniques, RNN and LSTM implementation, attention mechanism derivation, Transformer architecture mathematics, positional encoding, self-attention computation, multi-head attention implementation, training objectives for language modeling (causal and masked), gradient descent optimization, loss functions (cross-entropy), perplexity evaluation, computational complexity of Transformers, scaling laws, model efficiency techniques, bias and ethical considerations.

**iv)TEXTBOOKS**

1. Jurafsky, Daniel & Martin, James H, Speech and Language Processing (3rd Edition Draft), Pearson.(Comprehensive foundation in NLP, probabilistic language models, neural models, evaluation methods.)

2. Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron, Deep Learning, MIT Press, 2016.

(Mathematical foundations of neural networks, optimization, regularization, sequence models.)

3. Tunstall, Lewis; von Werra, Leandro; Wolf, Thomas, Natural Language Processing with Transformers, O'Reilly Media, 2022.(Transformer architecture concepts and implementation insights.)

**v) REFERENCES**

1. Vaswani, Ashish et al, Attention Is All You Need, NeurIPS, 2017.(Original Transformer architecture paper.)

2. Jay Alammar, The Illustrated Transformer (Online Technical Resource).(Conceptual visualization of self-attention and Transformer blocks.)

3. Andriy Burkov, The Hundred-Page Machine Learning Book, 2019.

(Concise overview of ML foundations useful for model training understanding.)

4. Raschka, Sebastian; Liu, Yuxi; Mirjalili, Vahid, Machine Learning with PyTorch and Scikit-Learn, Packt Publishing, 2022.(Hands-on neural network and deep learning implementation.)

**v) COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| I | **Module 1: Statistical Foundations of Language Modeling**<br><br>Probability theory refresher, Chain rule of probability, n-gram language models, Maximum likelihood estimation, Laplace smoothing, Perplexity, Limitations of statistical language models. | 5 |

| | | |
|---|---|---|
| II | **Module 2: Neural Language Models**<br><br>Feedforward neural language models, Word embeddings and vector space representation, Backpropagation derivation, Loss functions, Optimization algorithms (SGD, Adam), Implementation of simple neural language model using NumPy/PyTorch (without high-level wrappers). | 6 |
| III | **Module 3: Sequence Models and Attention**<br><br>RNN mathematical formulation, LSTM and GRU gating mechanisms, vanishing gradient problem, Attention mechanism derivation, Scaled dot-product attention, Mathematical interpretation of self-attention. | 7 |
| IV | **Module 4: Transformer Architecture from Scratch**<br><br>Encoder block components; multi-head attention implementation; Positional encoding derivation; Layer normalization; Residual connections; Decoder architecture; Causal masking; Complexity analysis $O(n^2)$. | 6 |
| V | **Module 5: Training a Mini Language Model**<br><br>Tokenization methods (character-level and subword implementation), Dataset preparation, Training loop design, Gradient clipping, Evaluation using perplexity, Model debugging, Ethical implications and model limitations. | 6 |
| | **Total Hours** | **30** |

**LAB / PRACTICAL**

| | Experiment | No. of hours |
|---|---|---|
| **Module-1** | | |
| 1 | a) Implement n-gram language model from scratch.<br>b) Compute probability of sentence<br>c) Calculate perplexity | 4 |
| **Module-2** | | |
| 2 | a) Build a feedforward neural language model.<br>b) Implement embedding layer manually.<br>c) Train using custom corpus | 4 |
| **Module-3** | | |
| 3 | a) Implement RNN and LSTM from scratch.<br>b) Implement scaled dot-product attention.<br>c) Compare RNN vs Attention performance. | 4 |
| **Module-4** | | |
| 4 | a) Implement complete Transformer block.<br>b) Build multi-head attention manually.<br>c) Implement positional encoding. | 4 |

| | | |
|---|---|---|
| **Module-5** | | |
| 5 | a) Train a small GPT-style causal language model.<br>b) Generate text from a trained model.<br>c) Evaluate using perplexity.<br>d) Analyze training instability. | 4 |
| **Project** | | |
| 6 | Students must:<br>a) Collect a domain-specific text corpus<br>b) Implement tokenizer<br>c) Design a mini-Transformer model<br>d) Train from scratch (no pretrained weights)<br>e) Evaluate performance<br>f) Document training curves and computational cost<br>g) Present limitations and ethical analysis | 10 |
| | **Total Hours** | **30** |

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26AC061A | **Research Methodology & IPR** | AC | 2 | 0 | 0 | 0 | 2026 |

**i) COURSE OBJECTIVES**

This course is intended to prepare the M. Tech students to carry out their dissertation/ research project work effectively, with a research bias. The student will be able to formulate a viable research problem, do a critical analysis of publications in the area of research, and identify a research method suitable for the work. The student will achieve the capability to write a technical paper based on his/her dissertation/ research project.

**ii) COURSE OUTCOMES:**

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

| # | Description | Level |
|---|---|---|
| CO1 | Explain research ethics, Citation, Impact factor and Plagiarism. | Apply |
| CO2 | Formulate a research problem, make a suitable research design, and identify the data collection methods. | Apply |
| CO3 | Analyze the collected data. | Analyze |
| CO4 | Explain the role of IPR and Patent law in fostering research work, leading to creation of improved products, thus supporting economic growth and social benefits. | Apply |
| CO5 | Write a technical paper for publication. | Apply |

**iii) SYLLABUS:**

Introduction to Research Methodology- motivation for research, types of research, ethical issues. Identifying a research area and collecting related literature. Research problem-scope-objectives, literature review, identifying research gaps, and formulating the research problem. Research design and methods, data collection and analysis. Copyright – royalty - IPR and patent law. Process of patenting and development, Procedure for grant of patents. Copy left- open access, citation, plagiarism, Impact factor. Writing a technical paper.

**iv) REFERENCES:**


1) Stuart Melville and Wayne Goddard, *Research methodology: an introduction for science & engineering students.*
2) Ranjit Kumar, 2nd Edition, *Research Methodology: A Step by Step Guide for beginners*.
3) Ramappa T*., Intellectual Property Rights Under WTO*, S. Chand, 2008.
4) Robert P. Merges, Peter S. Menell, Mark A. Lemley, *Intellectual Property in the New Technological Age*, 2016.
5) Mayall, *Industrial Design*, McGraw Hill, 1992. Niebel, "Product Design", McGraw Hill, 1974.


**v) COURSE PLAN:**

| **Module** | **Contents** | **Hours** |
|---|---|---|
| **I** | Introduction to Research Methodology: Motivation towards research, Types of research. Professional ethics in research: Ethical issues, ethical committees. Identification of major conferences and important journals in a chosen area of interest. Collection of at least 10 published papers on a research problem in the chosen area. | **6** |
| **II** | Defining and formulating the research problem: Literature Survey, Analysing the collected papers to understand how the authors have identified the research gaps, arrived at their objectives, and formulated their research problem. Understanding how their research work is different from the previous works in the chosen area. | **6** |
| **III** | Research design and methods: Analyzing the collected papers to understand how the authors have formulated the research methods, both analytical methods and experimental methods. Data Collection and analysis: Analyzing the collected papers to understand the methods of data collection, data processing, analysis strategies, and tools used for analyzing the data. | **6** |
| **IV** | Copy right - royalty - Intellectual property rights and patent law – Process of Patenting and Development, Procedure for grant of patents. Reproduction of published material: Copy left- Open access, Citation and acknowledgement. Plagiarism, Impact factor. | **6** |
| **V** | Technical writing - Structure and components of a typical technical paper, abstract and conclusion, illustrations and tables, bibliography, referencing and footnotes. Writing a technical paper – based on the identified research problem, and using the | **6** |

| | | |
|---|---|---|
| | collected papers, Literature survey, Problem formulation, and Research design, and a hypothetical result. | |
| | **Total hours** | **30** |

# PROJECT

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 26CS267A | **MINI PROJECT** | PR | 0 | 0 | 6 | **3** | 2026 |

**COURSE OBJECTIVES**

To make students

1) Collect the recent publications related to the identified Mini project.
2) Do a detailed study of the Mini project based on current journals, published papers and books.
3) Present a seminar based on the Mini project.
4) Improve the writing and presentation skills.
5) Design and develop a system or application in the area of their specialization.

**APPROACH**

1) Students shall make a presentation for 20-25 minutes based on the detailed study on the project and submit a report of the study.
2) There will be two interim progress reviews of the Mini project work. The first review will focus on the topic, objectives, methodology, design and expected results.
3) The second review shall focus on the work/ Implementation and results obtained.

**EXPECTED OUTCOME**

Upon successful completion of the Mini project and Seminar, the student should be able to

1) Identify and solve various problems associated with designing and implementing a system or application.
2) Test the designed system or application.
3) Improve the writing and presentation skills.
4) Explore domains of interest so as to pursue the course project

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 22CS278A | **DISSERTATION PHASE I** | PR | 0 | 0 | 24 | **16** | 2026 |

To make students

1) Do an original and independent study on the area of specialization.
2) Explore in depth a subject of his/her own choice.
3) Start the preliminary background studies towards the project by conducting literature surveys in the relevant field.
4) Broadly identify the area of the project work, familiarize with the tools required for the design and analysis of the project.
5) Plan the experimental platform, if any, required for project work.

**APPROACH**

1) There will be three interim progress reviews of the Project (Phase I). The first review shall focus on the topic, and objectives. This review will be conducted within one month of the commencement of third semester classes.
2) The second review shall focus on the methodology. This review will be conducted within two months of the commencement of third semester classes.
3) The third review shall focus on the design and expected results, and scope of the work which has to be accomplished in the fourth semester. This review will be conducted towards the close of the third semester.

**EXPECTED OUTCOME**

Upon successful completion of the Project (Phase I), the student should be able to

1) Identify the topic, objectives and methodology to carry out the project.
2) Finalize the project plan for their course project.

| Course Code | Course Name | Category | L | T | P | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|
| 22CS278B | **DISSERTATION PHASE II** | PR | 0 | 0 | 24 | **16** | 2026 |

**APPROACH**

To continue and complete the project work identified in Project (Phase I).

1) There will be three interim progress reviews of the Project (Phase II). The first review shall focus on the progress of the implementation of the design made in Project (Phase I). This review will be conducted within one month of the commencement of third semester classes.
2) The second review shall focus on the quality and quantum of the work completed. This review will be conducted within two months of the commencement of third semester classes.
3) The third review shall focus on the completed implementation and the results. This review will be conducted towards the close of the third semester.
4) At least one technical paper has to be prepared and published in journal conferences based on their project work.

## 2. Course types in this Curriculum structure

AC: Audit Course

IEC: Industry Elective Course - (offered by an industry)

LBC: Laboratory Course

PCC: Program Core Course

PCCP: Program Core Course with practical component

PEC: Program Elective Course

PECP: Program Elective Course with practical component

PR: Project Course - (Internship Course also belongs to this Course type)

SAEC: Skill/Ability Enhancement Course. This is a MOOC offered by AICTE/ NPTEL/ SWAYAM/ NITTTR)

## 3. Credit requirements for registering to higher semesters

| Semester | Allotted credits | Cumulative credits | Minimum credits required |
|---|---|---|---|
| M1 | 20 | 20 | Not Applicable |
| M2 | 18 | 38 | Not Insisted |
| M3 | 19 | 57 | 12 credits from M1 |
| M4 | 16 | 73 | Not Insisted |

1 hour of Lecture per week: 1 credit
2 hours of Mini project per week: 1 credit
1½ hours of Laboratory/ Project per week: 1 credit

## 4. Course Coding Scheme

**Structure of a Course Number: ABDCPYTN(R)**

| # | Character in the Course code | Description | Remarks |
|---|---|---|---|
| 1. | **AB** | Year of curriculum introduction | The last two digits of the year in which the curriculum is introduced |
| **2.** | **DC** | Department offering the course/ Category of the course<br><br>(The order for inclusion in the course code : MA, AC Dept code) | CE: Department of CE<br>CS: Department of CSE<br>EE: Department of EEE<br>EC: Department of ECE<br>ME: Department of ME<br><br>MA: Mathematics course<br>AC: Audit Course offered by the department/ from outside the department |
| **3.** | **P** | **P**rogram number<br><br>(In the chronological order of commencement of the M.Tech program in the department) | 1: Course offered for the 1st M.Tech program in the department<br>2: Course offered for the 2nd M.Tech program in the department<br><br>0: Course offered for more than one program/ courses offered from outside the department. |
| **4.** | **Y** | **Y**ear/ Level | 6: First/ Second semester M.Tech<br>7: Third/ Fourth semester M.Tech |

| | | | |
|---|---|---|---|
| **5.** | **T** | **T**ypes of Course | 1: Program Core course/ AC<br>2,3: Program Elective course<br>4,5: Skill/Ability Enhancement Course<br>6: Industry Elective course<br>7: Mini Project course<br>8: Project course/ Internship course<br>9: Laboratory course |
| **6.** | **N** | **N**umber for the course | A – Z shall be used for unique identification of the Course. |
| 7. | **(R)** | **R**evision number | (1): 1st revision<br>(2): 2nd revision etc.<br>:**Blank** at the introduction of the course |

| **Sample Course code** | **Description** |
|---|---|
| 26CE161B | A program core course in the 2026 curriculum offered in the first/ second semester of M.Tech in SE, introduced for the first time |
| 26CS162M(2) | A program elective course in the 2026 curriculum offered in the first/ second semester of M.Tech in CSE, in its second revision |

# 5. Assessment Pattern

## a) Program Core Course/ Program Core Course with practical component

A Program Core Course can be conducted as a theory course or a theory course along with its related laboratory experiments. A Program Core Course conducted as a theory course along with its related laboratory experiments comes under the course type PCCP. A PCC/ PCCP is evaluated out of 100 marks; 50 marks for Continuous internal assessment (CIA) and 50 marks for End semester evaluation (ESE).

Evaluation shall include application, analysis, and design based questions for both CIA and ESE.

### Continuous Internal Assessment (CIA): 50 marks

Micro project/Laboratory/ Course based project: 30 marks

Course based task/ Seminar/Quiz: 10 marks

Continuous Assessment Test (CAT), 1 No: 10 marks
(CAT shall include minimum 60% of the syllabus)

Micro project/ Course based project shall be done individually. Group projects are not permitted.

### End Semester Examination (ESE): 50 marks

ESE will be conducted by the Controller of Examinations (CoE). Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question. A question can have sub parts. Students shall answer any five questions.

The questions shall be useful in the testing of knowledge, skills, comprehension, application, analysis, synthesis, overall achievement and maturity of the students in a course, through questions relating to theoretical/ practical knowledge, derivations, problem solving and quantitative evaluation.

## b) Program Elective Course/ Program Elective Course with practical component

A Program Elective Course can be conducted as a theory course or a theory course along with its related laboratory experiments. A Program Elective Course conducted as a theory course along with its related laboratory experiments comes under the course type PECP. A PEC/ PECP is evaluated out of 100 marks; 50 marks for CIA and 50 marks for ESE.

Evaluation shall include application, analysis, and design based questions for both CIA and ESE.

**Continuous Internal Assessment: 50 marks**

| | |
|---|---|
| Preparing a review article based on peer reviewed original publications (Minimum 10 publications shall be referred)/ Micro project/Laboratory : | 30 marks |
| Course based task/ Seminar/ Data collection and interpretation: | 10 marks |
| Continuous Assessment Test (CAT), 1 No: (CAT shall include minimum 60% of the syllabus) | 10 marks |

**End Semester Examination: 50 marks**

The ESE will be conducted by the CoE. Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question. A question can have two or more sub parts. Students shall answer any five questions.

The questions shall be useful in the testing of knowledge, skills, comprehension, application, analysis, synthesis, testing of overall achievement and maturity of the students in a course, through questions relating to theoretical/ practical knowledge, derivations, problem solving and quantitative evaluation.

## c) Audit Course (Research Methodology and IPR)

An audit course is evaluated out of 100 marks; 50 marks for CIA and 50 marks for ESE.

**Continuous Internal Evaluation: 50 marks**

| | |
|---|---|
| Course based task: | 20 marks |
| Seminar/Quiz: | 20 marks |
| Continuous assessment Test (CAT), 1 No: (CAT shall include minimum 60% of the syllabus) | 10 marks |

**End Semester Examination: 50 marks**

The ESE will be conducted by the CoE. Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question. Students shall answer any five questions.

## d) Internship

Internships are educational and career development opportunities, providing practical experience in a field or discipline. They are structured, short-term, supervised placements often focused around particular tasks or projects with defined timescales. A student has the opportunity to do internship for one semester either in M3 or in M4. Such students will carry out a Project work in the other semester.

An internship may be compensated or non-compensated by the organization providing the internship. Internship has to be meaningful and mutually beneficial to the intern and the organization. It is important that the objectives of the internship program are clearly defined and understood. Internship offers the students an opportunity to

(i) Gain hands-on industrial or organizational exposure
(ii) Integrate the knowledge and skills acquired through his/her coursework
(iii) Interact with professionals and other interns, and
(iv) Improve the presentation, writing, and communication skills of interns.

Internship often acts as a gateway for final placement for many students.

A student shall carry out the Internship at an Industry/ Research Organization or at another institute of higher learning and repute (Academia). The students must select the organization for doing Internship on their own, with prior approval from the respective PG Programme coordinator. Every student shall be assigned a Faculty supervisor at the beginning of his/her Internship. The training shall be related to their specialization. The internship must be carried out for a duration of four to five months, during the third semester or fourth semester. On completion of the Internship course, the student is expected to be able to develop skills in facing and solving the problems experienced in the related field.

**Objectives**

- Exposure to the industrial environment, which cannot be simulated in the class room and hence creating competent professionals for the industry.
- Provide possible opportunities to learn, understand and sharpen the real time technical/ managerial skills required at the job.
- Exposure to the current technological developments relevant to the subject area of training.
- Create conducive conditions with quest for knowledge and its applicability on the job.
- Understand the social, environmental, economic and administrative considerations that influence the working environment.
- Exposure to the engineer’s responsibilities and ethics.

**Benefits of Internship to Students**

- An opportunity to get hired by the Industry/ organization.
- Practical experience in an organizational setting and Industry environment.

- An opportunity to see how the theoretical aspects learned in classes are integrated into the practical world. On-floor experience provides much more professional experience which is often worth more than classroom teaching.
- Helps the intern to decide if the industry and the profession is the best career option to pursue.
- Opportunity to learn new skills and supplement knowledge.
- Opportunity to practice communication and teamwork skills.
- Opportunity to learn strategies like time management, multi-tasking etc in an industrial setup.
- Makes a valuable addition to their resume
- Enhances their candidacy for higher education/placement.
- Creating networks and social circles and developing relationships with industry people.
- Provides opportunity to evaluate the organization before committing to a fulltime position.

**Benefits of Internship to the Institute**

- Build industry academia relations.
- Makes the placement process easier.
- Improve institutional credibility and branding.
- Curriculum revision can be made based on feedback from Industry/ students.
- Improvement in teaching learning process.

**Benefits of Internship to the Industry**

- Availability of ready to contribute candidates for employment.
- Students bring new perspectives to problem solving.
- Visibility of the organization is increased on campus.
- Quality candidate's availability for temporary or seasonal positions and projects.
- Freedom for industrial staff to pursue more creative projects.
- Availability of flexible, cost-effective workforce not requiring a long-term employer commitment.
- Cost-effective way to recruit and evaluate potential employees.
- Enhancement of employer's image in the community by contributing to the educational enterprise.

**Types of Internships**

- Industry Internship with/ without Stipend
- Government / PSU Internship (BARC/ Railway/ ISRO etc.)
- Internship with prominent education/ Research Institutes
- Internship with Incubation centers/ Start-ups

**Guidelines**

- All the students need to go for an internship for a minimum duration of four months and a maximum duration of five months.
- Students can take mini projects, assignments, case studies by discussing it with concerned authority from industry and can work on it during internship.

- All students should compulsorily follow the rules and regulations of the industry.
- Every student should take prior permissions from concerned industrial authority if they want to use any drawings, photographs or any other document from industry.
- Students should follow all ethical practices and Standard Operating Procedure (SOP) of the industry.
- Students must take necessary health and safety precautions as laid by the industry.
- Students should contact his /her Guide/Supervisor from the College on a weekly basis to communicate the progress.
- Each student has to maintain a diary/log book
- After completion of internship, students are required to submit
  - ✔ Report of work done
  - ✔ Copy of Internship certificate
  - ✔ Feedback from internship mentor in the place of internship
  - ✔ Proof of stipend (in case of paid internship).

**Evaluation of Internship**

Internship will be evaluated out of 100 marks for CIA.

| | |
|---|---|
| Student's diary/ Daily Log: | 25 Marks |
| Evaluation done by the Industry: | 25 Marks |
| Internship Report: | 25 Marks |
| Comprehensive Viva Voce: | 25 Marks |

**Student's Diary/ Daily Log**:

The main purpose of writing a daily diary is to cultivate the habit of documenting and to encourage the students to search for details. It develops the students' thought process and reasoning abilities. The students should record in the daily training diary the day to day account of the observations, impressions, information gathered and suggestions given, if any. It should contain the sketches and drawings related to the observations made by the students. A student's diary must be signed each day by the supervisor/ in charge of the section where the student has been working.

**<u>Format of  Student's Diary</u>**

Name of the Organization/Section:

Name and Address of the Section Head:

Name and Address of the Supervisor:

Name and address of the student:

Internship Duration: From ........................... To ............................................

Brief description about the nature of internship:

| Day | Brief write up about the Activities carried out: Such as design, sketches, result observed, issues identified, data recorded, etc. |
|---|---|
| 1 | |
| 2 | |
| 3 | |

Signature of Industry supervisor

Signature of Head/ HR Manager

Office Seal

**<u>Format of Attendance Sheet</u>**

Name of the Organization/ Section:

Name and Address of the Section Head:

Name and Address of the Supervisor:

Name and address of the student:

Internship Duration: From ........................... To ............................................

| Month & Year | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | … |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | | | | | |
| Month & Year | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |
| Month & Year | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | |

Signature of Industry supervisor Signature of Head/ HR Manager

Office Seal

**<u>Note:</u>**

- Student’s Diary shall be submitted by the students along with attendance record and an evaluation sheet duly signed and stamped by the industry to the Institute immediately after the completion of the training.
- The Attendance Sheet should remain affixed in the daily training diary. Do not remove or tear it off.
- Students shall sign in the attendance column. Do not mark ‘P’.
- Holidays should be marked in red ink in the attendance column. Absence should be marked as ‘A’ in red ink.

Student’s diary will be evaluated on the basis of the following criteria:

- Regularity in maintenance of the diary
- Adequacy and quality of information recorded
- Drawings, design, sketches and data recorded
- Thought process and recording techniques used
- Organization of the information.

**Format for Evaluation of Intern by industry**

Student Name : Date:
Supervisor Name : Designation:
Company/ Organization :
Internship Address:
Dates of Internship: From__________________ To________________________

***Please evaluate intern by indicating the frequency with which you observed the following parameters:***

| Parameters/ Marks | Needs improvement (0 – 0.25 marks) | Satisfactory (0.25 – 0.5 marks) | Good ( 0.75 marks) | Excellent (1 mark) |
|---|---|---|---|---|
| Behavior | | | | |
| Performs in a dependable manner | | | | |
| Cooperates with coworkers and supervisor | | | | |
| Shows interest in work | | | | |
| Learns quickly | | | | |
| Shows initiative | | | | |
| Produces high quality work | | | | |
| Accepts responsibility | | | | |
| Accepts criticism | | | | |
| Demonstrates organizational skills | | | | |
| Uses technical knowledge and expertise | | | | |
| Shows good judgment | | | | |
| Demonstrates creativity/ originality | | | | |
| Analyzes problems effectively | | | | |
| Is self reliant | | | | |
| Communicates well | | | | |
| Writes effectively | | | | |
| Has a professional attitude | | | | |
| Gives professional appearance | | | | |
| Is punctual | | | | |
| Uses time effectively | | | | |

Overall performance of student Intern (Please Tick one):

Needs improvement ( 0.50 marks)
Satisfactory (1.0 mark)
Good (1.5 marks)
Excellent (2.0 marks)

Additional comments, if any (2 marks):

*Signature of Industry Supervisor* *Signature of Section Head/HR Manager*

*Office Seal*

**Internship Report:**

After completion of the internship, the student should prepare a comprehensive report to indicate what he has observed and learnt in the training period and should be submitted to the faculty supervisor. The student should prepare the final report on the assigned topics. Diary/ daily log will also help to a great extent in writing the report since much of the information has already been incorporated by the student into the diary. The training report should be signed by the Internship supervisor, PG Programme Coordinator and Faculty mentor.

The Internship report will be evaluated on the basis of following criteria:

- Originality
- Adequacy and purposeful write-up
- Organization, format, drawings, sketches, style, language etc.
- Variety and relevance of learning experience
- Practical applications, relationships with basic theory and concepts taught in the course

**Comprehensive Viva Voce:**

Viva Voce will be done by a committee comprising Faculty Supervisor, PG Programme Coordinator, and one faculty member from a sister department. This committee shall evaluate the internship report also.

## e) Laboratory Courses

Laboratory courses will have only continuous internal assessment and carry 100 marks. Final assessment shall be done by two examiners; one examiner will be a senior faculty from the same department.

**Continuous internal assessment: 100 marks**

| | |
|---|---|
| Performance in regular laboratory experiments: | 70 marks |
| Final assessment/ laboratory test: | 30 marks |

## f) Industry Elective Course

Engineering students frequently aspire to work in areas and domains that are key topics in the industry. There are concerns by recruiters that skill sets of engineering students do not match with the Industry requirements, especially in the field of latest topics.

Industry knowledge aids in the bridge building process between academic institutions and industry. It also aids students in expanding their knowledge and innovating by allowing them to create something new. Core engineering courses provide students with a strong foundation. Evolving technology necessitates new methods and approaches to progress, prosperity, and the inculcation of problem-solving techniques. Industry knowledge will enable the students to deal with any scenario more effectively, thus fulfilling the current industry demands.

Rapid technological advancements have resulted in a massive revival in the way engineering works in the industry. Projects necessitate the integration of knowledge and abilities from a diverse variety of engineering specialties, with the barriers between them becoming increasingly blurred.

Students can choose courses offered by Industries that cover a wide range of highly relevant topics such as artificial intelligence, internet of things, big data, automation, and other relatable courses.

IEC will be evaluated out of 100 marks;  50 marks for CIA and 50 marks for ESE.

**Continuous internal assessment: 50 marks**

The continuous internal evaluation will be done by the expert in the Industry handling the course, and the coordinator from the college.

Micro project/ Course based project: 30 marks

Course based task/Seminar/Quiz: 10 marks

Continuous assessment Test (CAT), 1 No: 10 marks

(CAT shall include minimum 60% of the syllabus)

**End Semester Examination: 50 marks**

ESE will be conducted by the CoE using the question paper provided by the industry. Duration of the examination shall be 180 minutes.

The question paper will contain 7 questions with minimum one question from each module, having 10 marks for each question.  Students shall answer any five questions.  Evaluation of the answer scripts will be done by the expert in the Industry handling the course or the coordinator from the college under the expert's guidance.

## g) Skill/ Ability Enhancement Course

SAEC is an online MOOC of 12 weeks duration and shall be considered only if it is conducted by the agencies namely AICTE/ NPTEL/ SWAYAM/ NITTTR.  The course should have a proctored/ offline end semester examination. The students can do the SAEC credited in M3 according to their convenience from their first semester, but shall complete it by the third semester.  The list of courses is to be approved by the concerned Board of studies.  A course may be approved only if at least 70% of the course content matches with the area/ stream of study. The course shall not be considered if its content has more than 50% overlap with a core/ elective course in the concerned discipline or with an open elective.

A credit of 3 and a grade point of 10 will be awarded to all students whoever successfully completes the SAEC credited in M3.  Marks/ GPA awarded to the other SAEC shall be used for SGPA/CGPA computation.

## h) Mini Project

Mini projects help to strengthen the understanding of the fundamentals through application of theoretical concepts, and to boost their skills and widen the horizon of thinking. The ultimate aim of an engineering student is to resolve a problem by applying theoretical knowledge. Doing more projects enhances problem solving skills. The Mini project ensures preparedness of students to undertake their project work in M3 and M4.  Students should identify a topic of interest in consultation with his/her PG Programme Coordinator.  They should demonstrate the novelty of the project through the results and outputs.  This mini project work is assessed in three evaluations, two interim evaluations and a final evaluation.  The evaluations will be done by a committee comprising of Project Coordinator, Two senior faculty members in the department, and the student's Project Supervisor

Final evaluation will be conducted only if the Interim project report approved by the student's supervisor is submitted. The Plagiarism level in the report should be ≤ 25%, assessed based on the overall similarity index given by Turnitin licensed to the College.

Mini Project will be evaluated out of 100 marks under CIA, and has no ESE.

**a) First evaluation:**

**Evaluation committee: 20 marks**

Literature Survey: 7 marks

Objectives and Methodology: 7 marks

Clarity of presentation: 6 marks

**b) Second evaluation:**

**Evaluation committee: 20 marks**

Design: 7 marks

Implementation plan: 5 marks

Expected results: 8 marks

**c) Final evaluation: 60 marks**

**a) Supervisor/ Guide: 10 marks**

Log book and Regularity: 5 marks

Overall evaluation of the project work: 5 marks

**b) Evaluation committee: 50 marks**

Demonstration of functionality/ specifications: 20 marks

Level of completion: 5 marks

Clarity of presentation: 5 marks

Knowledge on the project work: 5 marks

Interim project report:

Technical content: 5 marks

Adequacy of references: 5 marks

Templates followed: 5 marks

## i) Project

The students must carry out the project work either in the college or in any CSIR/ industrial R&D organization/ any other reputed Institute which have facilities to carry out project work in the proposed area.

**Project work outside the College**:

For doing project work outside the college, the following conditions are to be met:

- They have successfully completed the course work prescribed in the approved curriculum up to the second semester.
- The student has to get prior approval from the DLAC.
- Students availing this facility should continue as regular students of the College.
- Facilities required for doing the project work shall be available in the Organization/ Industry. A certificate stating the time period for which the facilities shall be made available to the student, issued by a competent authority from the Organization/ Industry shall be submitted by the student along with the application.
- The student should have an external as well as an internal supervisor. The internal supervisor should belong to the college and the external supervisor shall be a Scientist or Engineer from the Institution/ Industry/ R&D organization with which the student proposes to do his project work. The external supervisor shall be with a minimum Post graduate degree in the related area.
- The MOOC must be completed as per the curriculum requirements:
- The student has to furnish his/her monthly progress as well as attendance report signed by the external supervisor and submit the same to the concerned internal supervisor.
- The external supervisor is to be preferably present during all the stages of evaluation of the project.

**Internship leading to Project:**

The students who, after completion of 6 to 8 weeks internship at some reputed organization, are allowed to continue their work as project for the third and fourth semester can do so after getting approval from the DLAC. Such students shall make a brief presentation regarding the work they propose to carry out before the DLAC for a detailed scrutiny and to resolve its suitability for accepting it as an M.Tech project. Once accepted, they will be permitted to complete their project in that organization (where they have successfully completed their internship) during their third semester and fourth semester.

**Project as part of Employment:**

Students may be permitted to discontinue the programme and take up a job, provided they have completed all the courses till the second semester (FE status students are not permitted) prescribed in the approved curriculum. The project work can be done during a later period either in the organization where they work if it has an R & D facility, or in the College. Such students shall submit an application with details (copy of employment offer, and the plan of completion of their project) to the Dean (PGSR) through the HoD. When the student plans to do the project work in the organization with R & D facility where they are employed, they shall submit a separate application with the following details:

- Name of R & D Organization/Industry
- Name and designation of an external supervisor from the proposed organization/ industry (a scientist or engineer with a minimum post graduate degree in the related area), along with his profile, and consent letter.
- Name and designation of a faculty member of the College as internal supervisor, and his/her consent letter.
- Letter from the competent authority from the Organization/ Industry granting permission to do the project work.

- Details of the proposed project work along with the work plan for completion of the project.

DLAC will scrutinize the proposal and forward it to CLAC for approval.

When a student does his project work along with the job in the organization (with R & D facility) where they are employed, the project work shall be completed in four semesters (two semesters of dissertation work along with the job may be considered as equivalent to one semester of dissertation work at the college). He should complete the M. Tech programme within four years from the date of admission as per the regulation. Extensions may be granted based on requests from the student and recommendation of the supervisors. The method of assessment of the project will be the same as in the case of regular students.

**Evaluation of Project (Phase I) in M3**

Project (Phase I) will be evaluated out of 100 marks under CIA, and has no ESE. There will be two evaluations (first evaluation and final evaluation). The assessment shall be done by the student's Project Supervisor, and a committee composed of Project Coordinator, two senior faculty members in the department, and the student's Project Supervisor. Project Coordinator shall enter the marks in the CoE portal.

Final evaluation will be conducted only if the student has submitted the Interim project report approved by the Supervisor, and Plagiarism level in the Interim project report is ≤25%.

1) **First evaluation: 30 marks**

**Project Supervisor: 10 marks**

i. Progress of work: 5 marks
(Literature Survey, Objectives, Methodology)

ii. Log book and Regularity: 5 marks

**Evaluation committee: 20 marks**

i. Topic, Objectives: 5 marks

ii. Methodology and Implementation plan for the work in M3: 10 marks

iii. Clarity in presentation: 5 marks

2) **Final evaluation:70 marks**

**Project Supervisor: 25 marks**

i. Progress of work: 15 marks

ii. Log book and Regularity: 5 marks

iii. Interim project report: 5 marks

**Evaluation committee:45 marks**

i. Demonstration of work completed: 15 marks

ii. Presentation and Viva voce: 10 marks

iii. Implementation plan of work in M4: 5 marks

iv. Interim project report: 15 marks
 Technical content: 10 marks
 Adequacy of references and
 Templates followed: 5 marks

**Evaluation of Project (Phase II) in M4/ Project in M3 or M4**

The evaluation of Project (Phase II) has CIA for 100 marks, and ESE for 100 marks. The continuous internal assessment is done under two evaluations (first evaluation and final evaluation), by the student's Project Supervisor, and a committee comprising of Project Coordinator, two senior faculty members in the department, and the student's Project Supervisor. Project Coordinator shall enter the marks in the CoE portal.

Final evaluation will be conducted only if the student has submitted the project report approved by the Supervisor, and Plagiarism level in the project report is ≤25%.

**Continuous internal assessment: 100 marks**

**1) First evaluation: 40 marks**

**Project Supervisor: 15 marks**

i. Progress of work: 10 marks
 (Experimentation and results)

ii. Log book and Regularity: 5 marks

**Evaluation committee: 25 marks**

i. Demonstration of work completed: 15 marks

ii. Presentation and Viva: 10 marks

**2) Final evaluation: 60 marks**

**Project Supervisor: 15 marks**

i. Progress of work: 10 marks
 (Quality and quantum of work)

ii. Project report: 5 marks

**Evaluation committee: 45 marks**

i. Demonstration of work completed: 10 marks

ii. Presentation and viva: 10 marks

iii. Project report: 10 marks
 Technical content: 5 marks
 Adequacy of references: 5 marks

iv. Paper publication: 15 marks
 (Published/accepted for publication in a journal/conference)

**End semester examination (Viva-voce examination): 100 marks**

The ESE will be done by a committee that comprises of the Project Coordinator, an external expert (from industry or research/academic institute), and the student's Project Supervisor

Each department must submit a panel of external experts to Dean (PGSR), as per the academic calendar. The minimum qualification requirement for an external examiner is M.Tech. The number of experts to be submitted is one more than number of students divided by 6 (rounded to the next integer). Honorarium for the external expert will be as fixed by the College.

The Project coordinator will enter the ESE marks in the CoE portal.

**Marks Distribution for Viva-voce examination**

i. Innovation & originality: 15 marks
(Introduction, Recent and related literature, Scope of the work, Objectives)

ii. Implementation and execution: 20 marks
(Methodology and work plan, Results and discussions, Quality of work done)

iii. Project Documentation: 20 marks
(Introduction, Problem Statement , Literature review, Methodology, Results and discussions, Conclusions, Future work, References)

iv. Presentation and Defense: 40 marks
(Clarity and effectiveness of presentation, Ability to explain the project objectives, Methodology and Findings, Handling questions and providing satisfactory answers)

v. Publication: 5 marks
(Published/accepted for publication in a journal/conference)

## 6. Teaching Assistanceship (TA)

All M.Tech students irrespective of their category of admission, shall undertake TA duties for a minimum duration as per the curriculum. Being a TA, the student will get an excellent opportunity to improve their expertise in the technical content of the course, enhance communication skills, obtain hands-on experience in handling the experiments in the laboratory and improve peer interactions.

Typical responsibilities of a TA include the following:
a) Facilitate a discussion session or tutorial for a theory course
b) Facilitate to assist the students for a laboratory course
c) Serve as a mentor for students, and act as the course web-master

TAs may be required to attend the instructor's lecture regularly. A TA shall not be employed as a substitute instructor, where the effect is to relieve the instructor of his/her teaching responsibilities. Students who are doing their project work outside the college are not required to do TA work during their third semester and fourth semester.

**Handling a tutorial session:**

(i) The TA must meet the instructor and understand his/her responsibilities well in advance, attend the lectures of the course for which the TA is a tutor, work out the solutions for all the tutorial problems self, approach the teacher if there is any discrepancy or need help in solving the tutorial problems, use reference text books, be innovative and express everything in English only.

(ii) The TA must try to lead the students to the correct solutions by providing appropriate hints rather than solving the entire problem by self, encourage questions from the students, lead the group to a discussion based on their questions, plan to ask them some questions, be friendly and open with the students, simultaneously being firm with them.

(iii) The TA must keep track of the progress of each student in his/her group, give a periodic feedback to the student about their progress, issue warnings if the student is consistently under-performing, report to the instructor if a particular student is consistently underperforming, pay special attention to slow-learners and be open to the feedback and comments from the students and faculty.

(iv) After the tutorial session the TA may be required to grade the tutorials/assignments/tests. Make sure to work out the solutions to the questions self, work out possible alternate solutions to the same question, and discuss the marking scheme with the instructor.

(v) Consult the instructor and ensure impartial approach to the students in their grading. Follow basic ethics.

**Handling a laboratory Session:**

(i) Meet the faculty – in- charge a few days in advance of the actual lab class and get the details of the experiment, get clarifications regarding all aspects of the experiment and the expectations, prepare by reading about the theoretical background of the experiment, know the physical concepts involved in the experiment, go to the laboratory and check out the condition of the equipment/instrumentation, perform the laboratory experiment at least once before the actual laboratory class, familiarize with safety/ security aspects of the experiment/equipment/laboratory, prepare an instruction sheet for the experiment in consultation with the faculty, and keep sufficient copies ready for distribution to students for their reference.

(ii) Verify condition of the equipment/set up about 30 minutes before the students arrive in the class and be ready with the hand outs, make brief introductory remarks about the experiment, its importance, its relevance to the theory they have studied in the class, ask the students suitable questions to know their level of preparation for the experiment, discuss how to interpret results, ask them comment on the results.

(iii) Correct/evaluate/grade the submitted reports after receiving suitable instructions from the faculty in charge, continue to interact with students if they have any clarifications regarding any aspect of the laboratory session, including course grading, Carefully observe instrument and human safety in laboratory class. Prepare simple questions for short oral quizzing during explanation of experiments enables active participation of students, facilitate attention, and provide feedback and formative assessment.

## 7. <u>Points to Remember</u>

1) Each department must conduct an awareness programme to all M.Tech students on day 1 regarding the curriculum and the regulation. There will be a common induction

programme on 'Universal Human Values' for students of all M.Tech streams during the first three days of Semester 1.

2) The departments must publish the list of MOOCs suitable to their programmes.
3) While choosing the Industry and the Industry electives, it should be ensured that the programme is relevant and updated in that discipline. The Industry expert handling the elective shall be a postgraduate degree holder. The evaluation procedure shall also be clearly explained to them.
4) The departments may invite the Industries/research organizations during the first semester and inform them about the internship that the students can undergo in M3/ M4. The possibility of doing their project work at the Industry shall also be explored. They should also be made aware about the evaluation procedure of the internship. They may also be informed that it is possible to continue an internship provided it leads to their project work. Proposals may be collected from them for allotting to students according to their fields of interest.
5) Make sure that all internal assessments and the end semester examinations to be conducted by the college are carried out as per the assessment procedure listed in the curriculum. Any dilution from the prescribed procedure shall be viewed seriously.
6) Teaching assistance shall be assigned to all students as per the curriculum. However, a TA shall not be employed as a substitute instructor, where the effect is to relieve the instructor of his or her teaching responsibilities (strictly prohibited by University Policy).
7) The possible TA responsibilities include the following:
    a) Facilitate a discussion session or tutorial for a theory course
    b) Facilitate to assist the students for a laboratory course
    c) Serve as a mentor for students, and act as the course web-master