GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Diploma Engineering
Level: Diploma
Branch: Electronics & Communication Engineering / Information & Communication Technology
Subject Code : DI05000051
Subject Name : AI & ML

w. e. f. Academic Year:	2024-25
Semester:	5
Category of the Course:	PEC-04

Prerequisite:	Students should have a basic understanding of programming concepts using Python, including variables, conditional statements, loops, and functions. Basic knowledge of data handling and simple mathematical concepts such as mean, graphs, and basic statistics will help students understand machine learning concepts effectively. Familiarity with fundamental computer applications and problem-solving techniques is also desirable for better understanding of Artificial Intelligence and Machine Learning applications.
Rationale:	Artificial Intelligence (AI) and Machine Learning (ML) are transforming modern industries by enabling machines to learn from data and perform intelligent tasks. Applications of AI and ML can be seen in various domains such as healthcare, finance, social media, automation, and smart systems. It is therefore important for diploma engineering students to understand the fundamental concepts and practical applications of these technologies. This course introduces students to the basic principles of Artificial Intelligence, machine learning techniques, neural networks, and natural language processing. The course focuses on **conceptual understanding along with practical implementation using Python libraries**, enabling students to analyse data, build simple machine learning models, and perform basic text processing tasks. The course also introduces students to **recent developments such as Generative AI and Large Language Models**, helping them understand modern AI tools and their responsible usage. The knowledge gained through this course will help students develop analytical skills and prepare them for further studies, industry applications, and emerging technology domains related to Artificial Intelligence and Machine Learning.

Course Outcome:
After Completion of the Course, Student will able to:

No	Course Outcomes	RBT Level
01	Explain fundamental concepts and applications of Artificial Intelligence and Machine Learning.	R, U

02	Apply basic machine learning algorithms to solve simple classification and prediction problems using Python.	U, A
03	Describe the architecture of Artificial Neural Networks and implement simple neural network models.	U, A
04	Apply Natural Language Processing techniques for text preprocessing and analysis using Python libraries.	U, A
05	Identify modern trends in Artificial Intelligence such as Generative AI, Large Language Models, and ethical considerations in AI systems.	U

Revised Bloom's Taxonomy (RBT)* ***R → Remember , U → Understand , A → Apply*

Teaching and Examination Scheme:

Teaching Scheme (in Hours)			Total Credits L+T+ (PR/2)	Assessment Pattern and Marks				Total Marks
				Theory		Practical		
L	T	PR	C	ESE (E)	PA(M)	PA(I)	ESE (V)	
3	0	2	4	70	30	20	30	150

Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	**1.1 Introduction to Artificial Intelligence** • Definition of Artificial Intelligence • Goals and importance of AI • Difference between Artificial Intelligence and Machine Learning • Examples of AI in daily life (Google Maps, Netflix recommendation, chatbots) **1.2 Applications of Artificial Intelligence** • AI in healthcare • AI in finance • AI in education • AI in manufacturing and automation • AI in social media platforms **1.3 AI Development Tools and Libraries** • Introduction to Python for AI • Overview of commonly used libraries: • NumPy	8	20

	• Pandas • Matplotlib • Scikit-learn **1.4 Data Handling Using Python Libraries** • Importing datasets (CSV, Excel files) using Pandas • Basic dataset operations: • Viewing data • Selecting columns • Basic statistical summary **1.5 Data Visualization** Using Matplotlib: • Line graph • Bar chart • Pie chart • Scatter plot • Histogram Using simple real-life or hypothetical datasets. **1.6 Ethical Considerations in AI** • Bias in AI systems • Privacy issues • Responsible use of AI Only **brief discussion**.		
2.	**2.1 Introduction to Machine Learning** •Definition of Machine Learning •Difference between traditional programming and machine learning •Basic workflow of machine learning o Data collection o Data preprocessing o Model training o Model testing **2.2 Well-Posed Machine Learning Problem** •Components of ML problem: o Task (T) o Experience (E) o Performance measure (P) •Example of well-posed ML problems: o Email spam detection	10	20

	○ Movie recommendation ○ House price prediction **2.3 Types of Machine Learning** • Supervised Learning • Unsupervised Learning Comparison of: • Input data • Output • Applications Examples of: • Classification • Regression • Clustering (concept only) **2.4 Supervised Learning Algorithms** **Classification Algorithms** • k-Nearest Neighbours (KNN) • Decision Tree (basic concept) **Regression Algorithm** • Linear Regression Applications and simple working principle of each algorithm. **2.5 Model Training and Testing** • Training dataset • Testing dataset • Train-test split concept **2.6 Model Evaluation Techniques** Basic evaluation concepts: • Accuracy • Confusion Matrix (simple interpretation) • Precision and Recall (basic understanding)		
3.	**3.1 Introduction to Neural Networks** • Motivation for Neural Networks • Biological neuron vs Artificial neuron • Applications of Neural Networks: ○ Image recognition ○ Speech recognition ○ Recommendation systems **3.2 Structure of Artificial Neural Networks** • Components of ANN:	10	23

	○ Input layer ○ Hidden layer ○ Output layer • Basic working of neural networks • Concept of weights and bias **3.3 Activation Functions** Purpose of activation functions in neural networks. Common activation functions: • Sigmoid function • ReLU (Rectified Linear Unit) • Tanh function Graphical representation and simple examples. **3.4 Types of Neural Network Architectures** Basic introduction to: • Single layer neural network • Multi-layer neural network Applications and simple working. **3.5 Building a Simple Neural Network Model** Using Python libraries such as: • Scikit-learn OR • Keras (basic introduction) Simple neural network example for classification. Example datasets: • Iris dataset • Digit recognition dataset		
4.	**4.1 Introduction to Natural Language Processing** • Definition of Natural Language Processing • Importance of NLP in AI • Applications of NLP: ○ Chatbots ○ Machine translation ○ Sentiment analysis ○ Voice assistants **4.2 NLP Processing Pipeline** Basic steps involved in NLP tasks: • Text collection • Text preprocessing	9	20

	• Feature extraction • Model building **4.3 Text Preprocessing Techniques** Introduction and purpose of preprocessing. Common preprocessing techniques: • Tokenization (word and sentence) • Stop word removal • Stemming • Lemmatization • Part-of-Speech (POS) tagging Using **NLTK library in Python**. **4.4 Text Representation Techniques** Methods to convert text into numerical form for machine learning. Basic techniques: • Bag of Words (BoW) • Term Frequency (TF) • TF-IDF (Term Frequency – Inverse Document Frequency) Simple examples and applications. **4.5 Simple NLP Applications** Demonstration of basic NLP tasks such as: • Sentiment analysis of text • Word frequency analysis • Keyword extraction from text Using Python libraries.		

5.	**5.1 Emerging Trends in Artificial Intelligence**
	Overview of modern developments in AI:
	• Generative AI
	• AI-powered assistants
	• Autonomous systems
	• AI in healthcare, finance, and education
	Short real-world case studies.
	
	5.2 Introduction to Generative AI
	• Concept of Generative AI
	• Difference between traditional AI and Generative AI
	• Examples of generative AI systems:
	  ○ Text generation
	  ○ Image generation
		8	17

	○ Code generation Applications in industry. **5.3 Large Language Models (LLMs)** Basic introduction to LLMs: • What are Large Language Models • Examples of LLM-based systems • Capabilities of modern AI chat systems: ○ answering questions ○ writing code ○ generating text Applications of LLMs in education, business, and software development. **5.4 Prompt Engineering** Introduction to prompt engineering: • What is a prompt • Importance of prompt design • Basic prompt techniques: ○ clear instructions ○ role prompting ○ step-by-step prompting • Types of prompts ○ Zero-shot prompting ○ One-shot prompting ○ Few-shot prompting ○ Instruction-based prompting ○ Role-based prompting ○ Step-by-step (Chain-of-thought) prompting ○ Creative prompting ○ Comparative prompting **5.5 Ethical and Responsible AI** Discussion on responsible use of AI: • Bias in AI systems • Data privacy concerns • Misuse of AI-generated content • Importance of ethical AI development		
	Total	**45**	**100**

Suggested Specification Table with Marks (Theory):

Distribution of Theory Marks (in %)					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
30	**40**	**30**	**0**	**0**	**0**

Where R: Remember; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create (as per Revised Bloom's Taxonomy)

References/Suggested Learning Resources:

(a) Books:

1. Introduction to Machine Learning with Python, by Andreas C. Müller and Sarah Guido, Publication: O'Reilly
2. Hands On Machine Learning with Scikit Learn, Keras, and TensorFlow by Aurélien Géron, Publication: O'Reilly
3. Machine Learning for Absolute Beginners by Oliver Theobald, Publication: Scatterplot Press
4. Natural Language Processing with Python by Steven Bird, Ewan klein & Edward Loper, Publication: O'Reilly

(b) Open source software and website:

a. https://www.geeksforgeeks.org/machine-learning/
b. https://www.geeksforgeeks.org/artificial-intelligence/
c. https://www.geeksforgeeks.org/natural-language-processing-nlp-tutorial/
d. https://www.tutorialspoint.com/machine_learning_with_python/index.htm
e. https://www.kaggle.com/learn
f. https://colab.research.google.com
g. https://scikit-learn.org/stable/tutorial/index.html
h. https://www.tensorflow.org/tutorials
i. https://keras.io/examples/
j. https://www.nltk.org/book/
k. https://spacy.io/usage
l. https://huggingface.co/learn

Suggested Course Practical List:

Sr. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1	a) Installation and setup of Python environment for AI development (Anaconda / Google Colab). b) Introduction to Pandas for data import and	I	02

Sr. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
	export (Excel, CSV etc.). c) Numerical Computing & Analytics with Python (NumPy, Matplotlib).		
2	Perform data handling using Pandas library. a) Import dataset from CSV file. b) Display dataset information and summary statistics. c) Select columns and rows from dataset. d) Perform basic statistical analysis.	I	02
3	Implement data visualization using Matplotlib library. a) Line graph b) Bar chart c) Pie chart d) Histogram e) Scatter plot	I	02
4	Perform data cleaning on dataset: a) Identify missing values b) Handle missing data using drop() and fillna() c) Replace values using mean/median	I	02
5	Apply feature scaling techniques: a) Min-Max Normalization b) Standardization using StandardScaler c) Compare scaled vs original data	I	02
6	Build a machine learning model that can learn from the measurements of iris flower whose species is known so that we can predict the species for a new iris flower. Use Iris dataset from Scikit-learn and implement k-Nearest Neighbors (KNN) classifier.	II	02
7	Implement Linear Regression algorithm using Scikit-learn to predict numerical values such as house price or student marks.	II	02
8	Implement Decision Tree classifier using Scikit-learn and compare the performance with KNN algorithm.	II	02
9	Evaluate performance of machine learning models using evaluation techniques. a) Accuracy b) Confusion Matrix c) Precision and Recall	II	02
10	Demonstrate train-test split concept:	II	02

Sr. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
	a) Split dataset into training and testing sets b) Train model with different ratios c) Compare performance		
11	Analyze KNN performance with different K values: a) Train model with K=1,3,5,7 b) Compare accuracy c) Plot accuracy vs K	II	02
12	Implement K-Means clustering: a) Apply K-Means algorithm b) Visualize clusters c) Interpret grouping of data	II	02
13	Implement following activation functions using Python to understand neural network behavior. a) ReLU b) Sigmoid c) Tanh (plot their graphs using Matplotlib).	III	02
14	Implement following feed forward neural network using Python programming: a) Single layer feed forward neural network. b) Multi-layer feed forward neural network using Scikit-learn / Keras.	III	02
15	Build a simple neural network model using Iris dataset for classification using Python libraries such as Keras or Scikit-learn.	III	02
16	Perform manual neuron computation: a) Calculate weighted sum b) Apply bias c) Pass through activation function (ReLU/Sigmoid)	III	02
17	Perform following text preprocessing tasks using NLTK library: a) Tokenization (word and sentence) b) Stop word removal c) Stemming d) Lemmatization e) Part-of-Speech tagging	IV	02
18	Implement following text representation techniques in NLP: a) TF-IDF (Term Frequency – Inverse Document Frequency) b) Bag of Words (BoW) c) Word frequency analysis	IV	02
19	Implement a simple NLP application such as sentiment analysis or keyword extraction using Python libraries.	IV	02

Sr. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
20	Generate word cloud from text data: a) Load text data b) Generate word cloud c) Interpret frequent terms	IV	02
21	Demonstrate Generative AI and Prompt Engineering using Large Language Models (LLMs). a) Design different prompts for the same problem. b) Compare responses generated by AI models. c) Analyse quality of generated results.	V	02
22	Generate images using AI tools: a) Provide text prompts b) Generate images c) Analyze output quality	V	02
	Total		44 Hrs Min 30

(Suggested) List of Laboratory/Learning Resources Required:

1. Hardware Requirements
   - Computer systems with minimum: Intel i5 (or equivalent) processor, 8 GB RAM (16 GB recommended), 256 GB SSD
   - Internet connectivity (for datasets, APIs, and cloud tools)
   - Optional (for advanced work): GPU-enabled systems (for deep learning tasks)
2. Software Requirements
   - Operating System: Windows / Linux / macOS
   - Programming Language: Python (preferred)
   - Development Environments: Jupyter Notebook / JupyterLab, Google Colab
   - Python Libraries: NumPy (numerical computing), Pandas (data manipulation), Matplotlib / Seaborn (data visualization), Scikit-learn (machine learning algorithms), TensorFlow / Keras (deep learning), NLTK / SpaCy (NLP tasks)
3. Datasets and Data Sources
   - Standard datasets: MNIST Dataset
   - Online repositories: Kaggle
   - UCI Machine Learning Repository

4. Tools and Platforms
   - Version Control: GitHub
   - Optional Deployment Tools: Streamlit (for simple web apps), Flask
5. Learning and Documentation Resources
   - Online documentation: Official Python and library documentation
   - MOOCs / Tutorials: Coursera, edX

Suggested Project List:

A suggestive list of micro-projects is given here. This has to match the competency and the COs. Similar micro-projects could be added by the concerned course teacher:

1. Iris Flower Classification: Develop a machine learning model that classifies iris flowers into different species based on petal and sepal measurements using classification algorithms.
2. House Price Prediction: Build a regression model to predict house prices based on features such as area, number of bedrooms, and location.
3. Student Performance Prediction: Develop a model to predict student performance (pass/fail or grade category) using factors such as attendance, study time, and previous grades.
4. Spam Email Detection: Build a machine learning classifier that identifies whether an email is spam or not using text classification techniques.
5. Movie Recommendation System: Develop a recommendation system that suggests movies to users based on their previous viewing preferences.Sentiment Analysis on Product Reviews: Build a model that analyzes product reviews and determines whether the sentiment is positive, negative, or neutral.
6. News Article Classification: Develop a model that classifies news articles into categories such as sports, technology, politics, and entertainment.
7. Handwritten Digit Recognition: Build a neural network model to recognize handwritten digits using a dataset such as MNIST.
8. Word Frequency and Keyword Extraction: Develop a program that analyzes a text document and identifies frequently occurring words and keywords using NLP techniques.

9. Chatbot for Frequently Asked Questions: Develop a simple chatbot that answers frequently asked questions using Natural Language Processing techniques.

Suggested Activities for Students:

Other than the classroom and laboratory learning, the following are the suggested student-related co-curricular activities which can be undertaken to accelerate the attainment of the various outcomes in this course. Students should conduct the following activities in groups and prepare reports of about 5 pages for each activity. They should also collect/record physical evidences for their portfolio which will be useful for placement interviews:

A) Explore different data repositories and register for machine learning competitions on platforms such as Kaggle.

B) Undertake micro-projects in teams related to machine learning, neural networks, or natural language processing.

C) Give a seminar presentation on recent developments in Artificial Intelligence such as Generative AI, Chatbots, or Autonomous Systems.

D) Collect data from social media platforms such as Snapchat, Facebook, or Twitter and perform sentiment analysis using NLP techniques.

E) Analyse a public dataset (Kaggle/UCI repository) and perform basic data analysis and visualization using Python libraries such as Pandas and Matplotlib.

* * * * * * *