

# CAP450:ARTIFICIAL INTELLIGENCE AND INTELLIGENT SYSTEMS

L:2 T:0 P:2 Credits:3

**Course Outcomes:** Through this course students should be able to

CO1 :: recall key principles of AI, ML, and Python programming, and identify their roles in real-world scenarios through use cases.

CO2 :: understand the techniques involved in data pre-processing, natural language processing, and neural networks, along with the challenges and opportunities they present.

CO3 :: apply the learned techniques to build, optimize, and deploy neural networks, convolutional neural networks, and recommendation systems for practical AI applications.

CO4 :: analyze the effectiveness of AI models through case studies, and design and implement AI solutions for chosen use cases, critically evaluating their impact and results.

## Unit I

**Introduction to Artificial Intelligence** : Introduction to AI and Intelligent Agents, Search Methods and Knowledge Representation, Use Cases of Artificial Intelligence, Role of Machine Learning Engineer, Machine Learning Tools & Packages

**Python for Data Analysis** : Python Data Structures, Python Programming Fundamentals, Python Packages, Working with NUMPY, Pandas, Data Visualization, Matplotlib and Seaborn

## Unit II

**Data Wrangling Techniques** : Introduction to Data pre-processing, Importing the Dataset, Handling Missing data, Splitting the data into Train and Test set, Feature Scaling

**Natural Language Processing** : Introduction to Natural Language Processing and NLTK, Bag of Words model, Natural Language Processing in Python, Sentiment analysis using Natural Language Processing

## Unit III

**Introduction to Neural Networks** : Neuron and Activation Function, Neural Networks working and learning, Gradient Descent, Stochastic Gradient Descent and Backpropagation

**Recurrent Neural Networks** : Introduction to Recurrent Neural Networks, The idea behind Recurrent Neural Networks, The Vanishing Gradient Problem, STMs and LSTM Variations Predicting Google stock prices using RNN, Evaluating, Improving, and Tuning the RNN

## Unit IV

**Convolutional Neural Networks** : Introduction to Convolutional Neural Networks, Convolution Operation, ReLU Layer, Pooling, Flattening and Full Connection Classification of images using CNN, Evaluating, Improving, and Tuning the CNN

## Unit V

**Build & Deploy an AI Application** : Introduction to different modes of Deployments, Working with Flask framework, Building an application with Flask Framework, Integrating Deep learning model with Web Application

## Unit VI

**Recommendation Systems** : Collaborative Recommender System, Content-Based Recommender System, Designing of Recommender System

**Case Studies** : Improving customer experience, Attrition Analysis for IBM, Movie Lens Dataset Analysis, Stock Market Data Analysis, Usecase1: Facebook Auto-Tagging/Face Recognition in Mobiles, Usecase2: Google Assistant, Usecase3: Youtube/Netflix/Amazon Recommendation Systems, Usecase4: Google Translate

## List of Practicals / Experiments:

### List of Practical

- Install Python and required libraries (NumPy, Pandas, Matplotlib, Seaborn). Create a simple Python script for basic data manipulation. Explore Jupyter Notebooks for interactive coding and visualization.
- Import a dataset using Pandas. Handle missing data and outliers. Split the dataset into training and testing sets. Perform feature scaling on numerical features.
- Use NLTK for text processing in Python. Implement a Bag of Words model for text representation. Perform sentiment analysis on a sample dataset using NLP techniques.

- Implement a simple neural network using TensorFlow or PyTorch. Understand the concept of neurons, activation functions, and backpropagation. Train the model on a small dataset and evaluate its performance.
- Build an RNN model for time series prediction. Address the vanishing gradient problem using LSTM. Evaluate, improve, and tune the RNN for better predictions.
- Implement a CNN for image classification. Understand convolutional operations, pooling, and flattening. Evaluate, improve, and tune the CNN for better accuracy.
- Set up a Flask project for AI application deployment. Create a simple web interface for user interaction. Integrate a pre-trained deep learning model with the Flask application
- Implement a collaborative recommender system using user-item interactions. Evaluate the recommendation system performance. Fine-tune parameters for better recommendations
- Design and implement a content-based recommendation algorithm. Use user preferences and content features for personalized recommendations. Evaluate the system's effectiveness
- Choose a use case (e.g., Facebook Auto-Tagging) from the syllabus. Implement the chosen use case using the acquired skills and knowledge. Analyze the results and present findings in a report

**Text Books:** 1. THE HUNDRED-PAGE MACHINE LEARNING BOOK by ANDRIY BURKOV, NIL

**References:** 1. MACHINE LEARNING by S SRIDHAR, M VIJAYALAKSHMI, OXFORD UNIVERSITY PRESS