Course Title: Deep Learning			
Course Code: UDS 304	**No. of credits: 4**	**L-T-P:** 30-15-30	**Learning hours: 60**
			L: Lectures**; T:** Tutorials**; P:** Practicals
Pre-requisite (if any): Machine Learning and Applications (UDS 301)			
Department: Natural and Applied Sciences			
Course Coordinator: Dr Adwitiya Sinha		**Course Instructor:**	
Contact Details:			
Course Type: Major		**Course Offered in:** Semester-6	

Course Description

This course introduces the core concepts, architectures, and applications of deep learning and neural networks for data-driven artificial intelligence solutions. It covers neural network fundamentals, optimization techniques, and advanced architectures such as CNNs and RNNs. Through hands-on programming using Python and tools like TensorFlow, Keras, and Optuna, students will gain practical experience in building, tuning, and interpreting deep learning models. By the end, students will be able to integrate theoretical understanding with real-world case studies to develop intelligent, scalable, and explainable AI systems.

Course Objectives

- Understand the fundamental principles and architectures of deep learning models.
- Design neural network and deep learning techniques to solve real-world problems.
- Implement and interpret deep learning models using modern Python frameworks.

Course Content

Module	Topic	L	T	P
1	**Introduction to Deep Learning and Neural Networks**			
	This module introduces the foundational principles and importance of deep learning in modern data-driven systems. Students will understand the biological inspiration, architecture, and learning process behind the neural networks. Overview of Artificial Intelligence and Machine Learning, Introduction to Deep Learning, Biological and Artificial Neurons, Perceptron and Multilayer Perceptron (MLP), Activation functions, Forward and Backpropagation neural networks, Regularization, hands-on with TensorFlow.	6	2	4
2	**Deep Learning Architectures and Optimization**			
	This module focuses on various deep learning architectures and optimization methods for improving model accuracy and generalization. The learners will explore techniques for training, tuning, and regularizing deep neural networks. Shallow and Deep Networks, Optimization algorithms (Adam, RMSProp, AdaGrad, Momentum), Dropout, Batch Normalization, Early Stopping,	8	3	6

	Hyperparameter tuning, Overfitting and underfitting, Gradient Descent, Loss functions, and evaluation metrics using TensorFlow, and hands-on model optimization using Optuna Python library.			
3	**Convolutional Neural Networks and Applications**			
	This module covers architectures for processing visual data, and the students will learn how CNNs handle image data. Convolutional Neural Network (CNN), filters, convolution operations, feature maps, and pooling layers, data augmentation techniques model generalization and performance evaluation. Practical sessions on basic image classification and optimization using Keras library.	8	5	10
4	**Recurrent Neural Networks and Applications**			
	This module covers advanced architectures for processing the sequential data. Students will learn how RNN-based models are used to capture temporal dependencies in sequences. Recurrent Neural Network (RNN), LSTM, GRU for handling long-term dependencies in sequence data, sequence-based prediction tasks with real-world applications. The unit also includes hands-on sessions using Keras Tuner for model optimization hyperparameters tuning.	8	5	10
	Total	**30**	**15**	**30**

Practical Sessions	Introductory hands-on lab session on TensorFlow in Python for building and training simple neural networks.	-	-	4
	Practical sessions on visualizing model performance and learning curves using Matplotlib, Seaborn, and Optuna.			6
	Python hands-on session on implementation optimization of CNN models using Optuna and Keras for assessing model performance.	-	-	10
	Hands-on with sequence-based data with Keras Tuner for RNN model accuracy and overall performance.	-	-	10
	Total Practical Sessions	**-**	**-**	**30**

Evaluation Criteria

- Minor Test: Written test [at the end of teaching of modules 1 and 2] -- 20%
- Major Test: Written test [end of semester, full syllabus, modules 1, 2, 3, and 4] -- 40%
- Practical Test: Practical test [including modules 2, 3 and 4] -- 20%
- Project-based learning: Project demonstration and report [at end of module 4] -- 20%

Learning Outcomes

By the end of the course, students will:

- understand real-world significance of deep learning and AI
  [Module 1 and 2; Minor Test]
- develop in-depth knowledge of architectures and optimization of deep learning
  [Module 2 and 3; Practical Test]
- gain insights on deep learning models and their performance evaluation
  [Module 1, 2, 3, and 4; Project-based learning and Major Test]

Pedagogical Approach

- The course will provide knowledge on the application and implementation of deep learning through classroom discussions, lectures, tutorials, and assessments.
- The course will enable students to use and deploy different techniques of deep learning models.

Reading Resources Recommended (* = suggested readings)

- * Goodfellow, Ian, Yoshua Bengio, & Aaron Courville. (2016). Deep Learning. MIT Press.
- Chollet, François. (2021). Deep Learning with Python (2nd Edition). Manning Publications.
- Aggarwal, Charu C. (2018). Neural Networks and Deep Learning: A Textbook. Springer.
- Ravichandiran, Sudharsan. (2019). Hands-On Deep Learning with TensorFlow 2.0. Packt Publishing.
- Patterson, Josh, & Gibson, Adam. (2017). Deep Learning: A Practitioner's Approach. O'Reilly Media.

Official Documentations of Tools/Libraries

- TensorFlow, URL: tensorflow.org
- Keras, URL: keras.io
- PyTorch, URL: pytorch.org/docs/stable
- Hugging Face Transformers, URL: huggingface.co/docs/transformers
- Optuna hyperparameter optimization framework, URL: optuna.readthedocs.io/en/stable
- Captum AI, URL: captum.ai/tutorials

Student Responsibilities: The students are required:

- To be regular and attentive in theory classes and lab sessions
- To come prepared with readings that would be given in the theory class
- To come prepared in lab with readings from respective theory concepts taught in the class
- To sincerely participate in the day-to-day lab activities
- To incorporate real-word datasets as case-studies in project-based learning
- To submit assignments and project works on or before the deadlines

Course Designed by:

- Dr Adwitiya Sinha, Associate Professor, Department of Natural and Applied Sciences, TERI School of Advanced Studies, New Delhi

Course Reviewers:

- Dr. Satish Chand, Professor, School of Computer & Systems Sciences, Jawaharlal Nehru University, Delhi
- Dr. Dharmveer Singh Rajpoot, Professor, School of Engineering and Technology, Central University of Punjab, Punjab.