COURSE DESCRIPTION FORM

INSTITUTION   FAST School of Computing, National University of Computer and Emerging Sciences, Islamabad

PROGRAM(S) TO BE EVALUATED BS-CS Spring 2025

Course Description

Course Code	AI-4009
Course Title	Generative AI
Credit Hours	3
Prerequisites by Course(s) and Topics	None
Assessment Instruments with Weights (homework, quizzes, midterms, final, programming assignments, lab work, etc.)	*(see table and explanation below)*

Assessment Item	Number	Weight (%)
Assignments	3	15
Quizzes	4	10
Project	1	10
Sessional-I	1	12.5
Sessional-II	1	12.5
Final Exam	1	40

Explanation of Assessment:
Every assignment and project deliverable will **have a 2-hour grace period** during which no penalty will apply. This is intended to allow you time to verify that your submission has been submitted properly. We recommend you re-download it and look it over to make sure all deliverables have been answered. Deliverables after the grace period will receive a grade of 0.

Course Instructors	Dr. Akhtar Jamil
Lab Instructors (if any)	None
Course Coordinator	Dr. Akhtar Jamil
URL (if any)	
Current Catalog Description	This course provides an in-depth study of modern generative artificial intelligence, beginning with foundational concepts such as Bayesian learning, maximum likelihood estimation, and neural network architectures. Students will explore diverse model families including autoregressive models, autoencoders, Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), and diffusion models for image and data generation. The course also covers transformers and large language models (BERT, GPT, DeepSeek, Gemini) alongside multimodal systems capable of integrating text, vision, and

	other modalities. Advanced topics include prompt engineering, Retrieval-Augmented Generation (RAG) with vector databases, and multi-agent system frameworks such as the OpenAI Agents SDK, Google Agent Development Kit, LangGraph, Microsoft AutoGen, CrewAI, and LlamaIndex Workflows. Optimization strategies like parameter-efficient fine-tuning (LoRA, PEFT) and knowledge distillation are also included. With a balance of theory and practical implementation, the course enables students to design, evaluate, and build innovative generative AI solutions across text, image, and multimodal domains.
Textbook (or **Laboratory Manual** for Laboratory Courses)	
Reference Material	• Research papers from top conferences like NeurIPS, ICML, and CVPR. • Deep Learning, by Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press. • Neural Networks and Deep Learning : A Textbook, Charu C. Aggarwal • Deep Learning with Python, François Cholle • Machine Learning and Deep Learning with Python, scikit-learn and TensorFlow, Sebastian Raschka and Vahid Mirjalili
Course Learning Outcomes	**A. Course Learning Outcomes (CLOs)** After completion of the course, the students shall be able to: **1. CLO 1 (Understand – Level 2)** Explain and interpret the foundational principles of generative AI, including Bayesian learning, neural architectures (CNNs, RNNs, Transformers), and their role in generative modeling. **2. CLO 2 (Apply – Level 3)** Implement and experiment with core generative models such as autoregressive models, Autoencoders, VAEs, GANs, and Diffusion Models for data generation tasks. **3. CLO 3 (Analyze – Level 4)** Analyze and compare the architectures and training dynamics of different generative models (GAN families, VAEs, Diffusion, Transformers, LLMs), identifying their strengths, limitations, and appropriate application domains. **4. CLO 4 (Evaluate – Level 5)** Evaluate generative AI systems using established metrics (e.g., FID, BLEU, perplexity, CLIPScore), conduct comparative experiments, and interpret results to assess model performance and reliability. **5. CLO 5 (Create – Level 6)** Design and develop innovative generative AI solutions by integrating advanced techniques such as prompt engineering, Retrieval-Augmented Generation (RAG), multi-agent frameworks (OpenAI SDK, Google ADK, LangGraph, AutoGen, CrewAI, LlamaIndex Workflows), and optimization strategies (LoRA, PEFT, knowledge distillation) to solve real-world problems.

B.	Program Learning Outcomes	
	For each attribute below, indicate whether this attribute is covered in this course or not. Leave the cell blank if the enablement is little or non-existent.	
1. Academic Education:	To prepare graduates as computing professionals	✓
2. Knowledge for Solving Computing Problems:	Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.	✓
3. Problem Analysis:	Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.	✓
4. Design/ Development of Solutions:	Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.	✓
5. Modern Tool Usage:	Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.	✓
6. Individual and Team Work:	Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings.	✓
7. Communication:	Communicate effectively with the computing community and with society at large about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.	✓
8. Computing Professionalism and Society:	Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.	

9. Ethics:	Understand and commit to professional ethics, responsibilities, and norms of professional computing practice.	
10. Life-long Learning:	Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.	✓
11. Project Management and Finance	Demonstrate knowledge and understanding of management principles and economic decision making and apply these to one's own work as a member or a team.	
12. Life Long Learning	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological changes.	

C. Mapping of CLOs on PLOs
(CLO: Course Learning Outcome, PLOs: Program Learning Outcomes)

		PLOs											
		1	2	3	4	5	6	7	8	9	10	11	12
CLOs	1	✓	✓	✓	✓								
	2		✓	✓	✓	✓							
	3		✓	✓	✓			✓					
	4			✓	✓	✓		✓			✓		
	5		✓	✓	✓	✓	✓	✓					

Topics Covered in the Course, with Number of Lectures on Each Topic (assume 15-week instruction and one-hour lectures)

Week	List of Topics	Hrs	CLO(s)	Comments
1.	-Introduction to Course - Overview of Bayesian Learning - Maximum Likelihood Estimation	3	1	
2.	- Artificial Neural Networks - Convolutional Neural Networks (CNNs)	3	1	
3.	- Introduction to Autoregressive Models - PixelRNN and PixelCNN	3	2	Assignment #1
4.	- Latent Variable Models - Introduction to Autoencoders - Variational Autoencoders (VAEs)	3	2	
5.	- Introduction to Generative Adversarial Networks (GANs), - Deep Convolutional GAN (DCGAN)	3	2	Select Project Topic

6.	Sessional -I				
7.	- Conditional Image Generation using GAN (cGAN), - CycleGAN, - StyleGAN	3	2, 3	Assignment #2
8.	- Introduction to Diffusion Models - Latent Diffusion Models (LDM)	3	2, 3	
9.	- Introduction to Transformers and Vision Transformers	3	1,3	
10.	Introduction to Large Language Models, BERT	3	1, 3	Assignment #3
11.	Introduction to Large Language Models, GPT, DeepSeek v3, Gemini	3	3	
12.	Sessional – II				
13.	- Prompt Engineering - Retrieval Augmented Generation (RAG), FAISS or Weaviate or Pinecone	3	5	
14.	Introduction to Multi-Agent Systems, OpenAI Agents SDK, Agent Development Kit (Google), LangGraph (LangChain),	3	5	
15.	Model Context Protocol (MCP), Microsoft AutoGen, CrewAI, LlamaIndex Workflows	3	5	
16.	- Optimization: Fine Tuning LLMs (Low-Rank Adaptation, Parameter-efficient Tuning) - Knowledge Distillation	3	5	Project Submission
	Total	48		

Laboratory Projects/Experiments Done in the Course	Multiple architectures' implementation will be discussed in class and students will be required to do one project in which a paper will be implemented
Programming Assignments Done in the Course	Implementation different architectures to be done by students.

Class Time Spent (in hours)	Theory	Problem Analysis	Solution Design	Social and Ethical Issues
	26	9	10	3

Oral and Written Communications	Every student is required to submit at least __3___ written reports of typically _3____ pages and to make __1___ oral presentation of typically ____10___ minutes' duration.