Maharaja Surajmal Institute of Technology, New Delhi-Department of Computer Science and Engineering

Report on 3 days Lecture series

"Generative Artificial Intelligence"

Name of the Expert	Dr. Anil Sharma – Al Research Engineer at Samsung		
Date and Time	13-10-2025, 14-10-2025, 15-10-2025 (10:00 AM - 12:00 PM)		
Target Audience	3 rd Year Students		
Organized by	Department of Computer Science and Engineering, Maharaja Surajmal Institute of Technology		
Attended Participants	50		

1. Introduction

The 3-day lecture series on Generative AI at MSIT was aimed at equipping participants with a thorough understanding of generative models, their theoretical foundations, and practical applications. Over the three days, we explored fundamental AI/ML concepts, advanced generative architectures like GANs, conditional GANs (e.g., Pix2Pix), and modern AI techniques such as prompt engineering and retrieval-augmented generation (RAG). The workshop struck a careful balance between theory and hands-on practice, giving students a chance to implement models and witness their behaviour during training.

2. Objectives

The main goals of the lecture series were:

- 1. To provide a historical and conceptual foundation of artificial intelligence, tracing its evolution up to modern generative models.
- 2. To deeply understand generative adversarial networks (GANs), including their architecture, training dynamics, and challenges.
- 3. To familiarize students with conditional GANs (like Pix2Pix) and tasks such as image-to-image translation.

- 4. To introduce prompt engineering, covering techniques like zero-shot, few-shot, chain-of-thought, and more advanced methods.
- 5. To explain Retrieval-Augmented Generation (RAG) and how external data retrieval boosts LLM responses.
- 6. To build a working agentic AI model integrating prompt engineering + data retrieval + model inference.

3. Detailed lecture series Content

Day 1 — History of Al, ML Concepts & GAN Basics

a) History of AI & ML:

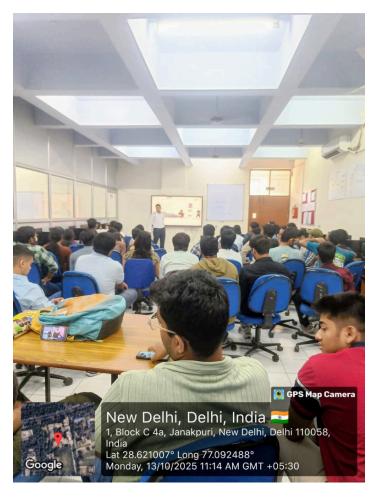
- We began by discussing the evolution of artificial intelligence: from early symbolic AI and rule-based systems to statistical machine learning, and eventually, deep learning and generative models.
- Key milestones included the invention of the transformer architecture (notably the "Attention Is All You Need" paper in 2017), which underpins many modern LLMs.
- We talked about how the paradigm shifted from purely discriminative models (for tasks like classification) to generative models that can create new data (images, text, etc.).

b) Machine Learning Fundamentals:

- Concepts such as supervised vs unsupervised learning, training and testing splits, overfitting, generalization, and optimization techniques were covered.
- We also discussed neural network basics: layers, activation functions, loss functions, and backpropagation.

c) Generative Adversarial Networks (GANs):

- We learned what a GAN is: a framework with two competing networks a Generator (G) and a Discriminator (D).
- The training process is adversarial: the generator tries to produce data that fools the discriminator, while the discriminator tries to distinguish real vs fake samples.
- We studied common issues in GAN training: mode collapse (generator produces limited diversity), instability, and how to monitor loss values.







Day 2 — Advanced Generative Models & Hands-on Training

a) Deep Convolutional GANs (DCGANs):

- We discussed DCGANs, a popular variant that uses deep convolutional layers (both in G and D) and batch normalization to improve stability and quality of generated outputs.
- The architecture helps in generating higher-quality images compared to simpler GANs.

b) Pix2Pix (Conditional GAN):

- We learned about conditional GANs (cGANs), where the generation is conditioned on some input, e.g., an image.
- Specifically, Pix2Pix uses a U-Net architecture for the generator (encoder-decoder with skip connections) and a PatchGAN discriminator.
 - o In Pix2Pix, skip connections (from encoder to decoder) help preserve fine-grained spatial detail in the output.
 - The discriminator doesn't classify the whole image as real or fake; instead, it makes decisions on small patches (e.g., 70×7070 \times 7070×70 patches).
 - Loss function: it combines adversarial loss (to fool the discriminator) + L1 reconstruction loss (difference between generated and real images), weighted appropriately.
- We learned that Pix2Pix performs well on paired image translation tasks (e.g., sketch → real image, map → photo) because training requires paired data.

c) Hands-on Training:

- In the practical session, we trained a GAN model on a dataset (the exact dataset was provided by the instructor).
- We observed how loss curves evolve for both generator and discriminator, how to adjust learning rate, and how batch size affects stability.
- We tried to optimize hyperparameters (learning rate, epochs, optimizer) to reduce common problems like oscillation or mode collapse.



Day 3 — Prompt Engineering, RAG & Building an AI Agent

a) Prompt Engineering:

- We covered several prompting techniques:
 - Zero-shot prompting: Giving instructions without examples.
 - Few-shot prompting: Providing a few input-output examples to guide the model.
 - Chain-of-Thought (CoT) prompting: Asking the model to reason step-by-step, helping with multi-step logical or arithmetic tasks.
 - Advanced techniques: Like Tree-of-Thought (ToT), where the model explores multiple reasoning paths.
- We also learned how to structure prompts for RAG-based systems:
 - When using RAG, it's important to instruct the LLM to rely on the retrieved documents and use only relevant context.
 - We practiced designing system instructions, context windows, and re-ranking retrieved documents.

b) Retrieval-Augmented Generation (RAG):

- We discussed what RAG is: a hybrid framework where a retriever module fetches relevant documents, and a generator module (LLM) uses that information to generate more accurate and context-aware responses.
- Advantages we studied:
 - Reduces hallucinations by grounding responses in real data.
 - Helps when working on domain-specific or up-to-date information not fully covered by the LLM's training data.
- We also saw practical prompt strategies for RAG, like how to tell the model to "consult" retrieved documents before answering.

c) Building the Agentic Model:

- In the final hands-on session, we combined prompt engineering + RAG to build a simple AI agent:
 - The agent could retrieve relevant documents or data (given a small dataset or knowledge base).
 - o It could then generate informed responses or analysis, using the LLM with the designed prompt.
 - We also tuned the prompt structure and retrieval strategy to make this agent more reliable.







4. Technical Concepts Explained

1. GAN Working Mechanism:

- The generator (G) takes as input a random noise vector sampled from a latent space (e.g., Gaussian) and outputs a fake sample (e.g., an image).
- The discriminator (D) receives both real data samples and fake samples generated by G, and outputs a scalar probability (between 0 and 1) representing its belief that the input is real.
- o The training objective is a minimax game:

$$\operatorname{Min}_D\operatorname{Max}_G\mathbb{E}_{real}log D(x)dx + \mathbb{E}_{fake}(1-D(G(z)))dz.$$

• This adversarial setup forces G to improve until it can fool D, and D to become better at distinguishing real vs fake.

1. Pix2Pix Architecture:

- Generator: U-Net encoder-decoder with skip connections to preserve spatial structure.
- Discriminator: PatchGAN classifies small patches (e.g., 70×70) rather than the whole image, which helps enforce more local realism.
- o Loss Function: The total generator loss = adversarial loss + $\lambda \cdot L1$ \lambda \cdot L1 $\lambda \cdot L1$ loss, where L1 loss encourages generated output to be close to the target image. In Pix2Pix, λ \lambda λ is often set to a high value to enforce reconstruction.
- Training details: In many implementations, optimizers like Adam with learning rate ~0.0002 are used.

2. Prompt Engineering Techniques:

- Zero-shot: Use only instructions, no examples.
- o Few-shot: Provide a small number of input-output pairs.
- o Chain-of-Thought (CoT): Ask the model to "think step by step," enabling reasoning.
- Tree-of-Thought (ToT): Extend CoT by asking the model to explore multiple reasoning paths and compare them.

3. Retrieval-Augmented Generation (RAG):

- In RAG, the model retrieves relevant documents from a knowledge base or vector store and uses them as context for generation.
- Proper prompt engineering for RAG is crucial: specify in prompt to "use only the retrieved context," manage context window, and choose relevant documents.
- Use cases: up-to-date Q&A systems, domain-specific chatbots, and tasks requiring factual accuracy.

5. Outcomes & Learnings

By the end of lecture series

• Conceptual Understanding: We gained a strong grasp of generative models (GANs, DcGANs), how they are trained, and their real-world applications.

- Hands-on Experience: We actually trained a GAN model, saw training dynamics, and understood practical challenges like unstable training or mode collapse.
- Model Implementation: Learning Pix2Pix gave us hands-on experience on image-to-image translation tasks.
- Advanced Prompting Skills: Through prompt engineering, we learned how to make LLMs reason better and produce more accurate outputs.
- RAG Knowledge: Understanding how retrieval and generation combine to produce more factual, grounded AI-generated responses.
- Agent Building: We built a mini-agent that analyzes data combining retrieval + generation which is very relevant for building AI systems in real applications.

6. Challenges & Limitations Observed

During the lecture series, we encountered and discussed some common challenges in generative AI:

1. Stability of Training:

- GANs are notoriously hard to train the balance between the generator and discriminator is delicate.
- o If the discriminator gets too strong, generator fails; if G is too good, D becomes useless.

2. Mode Collapse:

o Sometimes the generator produces very similar outputs (lacks diversity).

3. Data Requirements:

- Pix2Pix requires paired and well-aligned datasets, which may not always be available for certain real-world tasks.
- 4. Prompt Engineering Complexity:
 - Crafting prompts that consistently produce high-quality results requires experimentation.
 - Ensuring that RAG-based models rely correctly on retrieved context (and do not hallucinate) is non-trivial.

7. Future Scope & Recommendations

Based on our learnings, the following are recommended next steps and potential future applications:

• Project Ideas:

- 1. Develop an image-to-image translation application (e.g., sketch to photo, map to satellite, summer-to-winter landscapes) using Pix2Pix.
- 2. Build a domain-specific QA chatbot using RAG + prompt engineering, e.g., for academic research, customer support, or medical information.
- 3. Create an AI agent that uses retrieval + generation + reasoning (via CoT or ToT) for multi-step tasks (like planning, summarization, or decision support).
- Improvements in lecture series Content:
 - o Introduce Wasserstein GAN (WGAN) or other stable GAN variants in advanced sessions. WGAN addresses stability issues and mode collapse by using Wasserstein

distance.

- o Provide a larger, more diverse dataset to train GANs in real-world scenarios.
- o Include evaluation metrics for generative models: Inception Score (IS), Frechet Inception Distance (FID), etc., to measure quality of generated outputs.
- Dive deeper into multi-agent systems combining RAG, prompt chaining, and retrieval for complex workflows.

8. Conclusion

The 3-Day Generative AI lecture series provided a holistic learning experience, combining the history, theory, and technical depth of generative models with practical hands-on sessions. We significantly strengthened our understanding of GANs, conditional generative networks, and modern LLM-based techniques like prompt engineering and RAG.

This lecture series has not only enhanced our theoretical knowledge but also given us the practical skills to build and experiment with generative AI systems. Going forward, the insights and experience gained will be extremely valuable for both academic projects and potential real-world applications in AI.

9. Acknowledgements

We extend our sincere gratitude to Dr. Anil Sharma Sir for conducting the 3-day lecture series and sharing his valuable expertise, guidance, and practical insights on Generative AI. We also thank the Department of Computer Science & Engineering, MSIT, and all faculty members involved for organizing this workshop, providing students with an opportunity to gain hands-on experience, deepen their understanding of advanced AI concepts, and plan their academic and project work effectively in this emerging field.

Faculty Co-ordinator:

Dr. Kavita Sheoran

Dr. Sapna Malik

Dr. Medhavi Malik

Dr. Geetika Dhand Dr. Nishtha Jatana

HOD CSE (1 st Shift), HOD CSE (2nd Shift)