gen ai (notes)

What is Generative AI? (8 minutes)

- **Definition**: Generative AI refers to models that learn data distributions and can gener**ate new data points** from those distributions.

Formal Idea:
If data comes from some distribution $p_{data}(x)$, generative models try to approximate it with $p_{model}(x)$.
Once trained, the model can generate new $x' \sim p_{model}(x)$ that looks like the original data.

Types of Generative Models:

1. **Autoregressive Models** – generate one element at a time (e.g., GPT).
2. **Variational Autoencoders (VAEs)** – learn latent space + reconstruction.
3. **Generative Adversarial Networks (GANs)** – generator vs discriminator.
4. **Diffusion Models** – gradually remove noise to generate high-quality samples (e.g., Stable Diffusion).

Analogy:

- Predictive AI = "finish the exam by circling the right option."
- Generative AI = "write your own exam paper from scratch."

2. Core Technical Concepts (20 minutes)

This is the heart of the lecture — let's slow down here and explain **how generative AI really works**.

2.1 Neural Networks Refresher

At the foundation of GenAI are **neural networks**.

- Each neuron takes an input, multiplies by weights, adds a bias, and applies an activation function:

$$y = f(Wx + b)$$

- Layers of these neurons form **deep neural networks**.
- In GenAI, these networks are used to approximate very complex functions — like "what word is most likely next" or "what an image looks like."

Analogy: Think of it as a factory assembly line: each layer transforms the data a little bit, until at the end, you get the output.

2.2 Language Modeling (Autoregressive Models)

Generative text models like GPT use **autoregressive modeling**.

- The idea is simple: **predict the next token** given previous tokens.

$$P(x_1, x_2, \ldots, x_n) = \prod_{t=1}^{n} P(x_t \mid x_{<t})$$

- Example: Suppose the input is **"The cat sat on the"**.
  - The model calculates probabilities for the next word.
  - Maybe: *mat* (0.7), *chair* (0.2), *roof* (0.1).
  - It samples one, often choosing the highest probability, but sometimes picking others for creativity.

This step-by-step generation continues until the model finishes a sentence.

2.3 Transformer Architecture (The Backbone of Modern GenAI)

The **transformer** is the architecture that powers GPT, Claude, LLaMA, and Gemini.

Problem with older models

- RNNs and LSTMs read sequences word by word.
- They struggled with **long-range dependencies**. Example: *"The balloon that the boy was holding popped."* Understanding what popped (the balloon) requires remembering context far back.

Attention Mechanism

Transformers solved this using **self-attention**.

- Attention answers: *"Which other words in the sentence should I focus on when processing this word?"*
- Formula:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Here:

- **Q (Query)** = current word.
- **K (Key)** = other words.
- **V (Value)** = information about those words.

The model computes similarity between Q and K, and uses it to weight V.

Example: In *"The cat sat on the mat because it was tired"*, when processing *"it"*, attention helps the model focus on *"cat"*, not *"mat"*.

Transformer Layers

- Stacked attention + feedforward layers.
- Uses **multi-head attention** → multiple perspectives at once.

- Adds **positional encodings** so the model knows word order.

This allows models to read entire sequences at once and capture both local and global context efficiently.

2.4 Variational Autoencoders (VAEs)

Now let's move to models that generate images or structured data.

Idea

- VAEs learn to compress input into a **latent space** (like a summary) and then reconstruct it.
- During generation, we can sample from this latent space and decode new outputs.

How It Works

1. **Encoder**: Maps input xxx (say, an image) into a latent variable zzz.
2. **Decoder**: Reconstructs the input from zzz.
3. **Loss function**: Balances two goals:
    - Reconstruct the data well.
    - Make latent space continuous and smooth (so sampling works).

$$\mathcal{L} = \mathbb{E}_{q(z|x)}[\log p(x|z)] - KL(q(z|x)||p(z))$$

- The KL term ensures latent variables follow a Gaussian distribution, making it possible to sample.

Example: Train on handwritten digits → you can sample from latent space to generate *new* digits that look real but weren't in the dataset.

2.5 Generative Adversarial Networks (GANs)

GANs are another powerful family for image generation.

Idea

GANs work like a game between two networks:

- **Generator (G):** tries to create fake data that looks real.
- **Discriminator (D):** tries to tell apart real vs fake.

Training is like a **cat-and-mouse game**:

- G improves to fool D.
- D improves to detect G.

Objective

$$\min_G \max_D V(D,G) = \mathbb{E}_{x \sim p_{data}}[\log D(x)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))]$$

Example

- Train on celebrity faces → Generator learns to make *new* realistic faces.
- Issues: Mode collapse (producing similar samples), training instability.

2.6 Diffusion Models (e.g., Stable Diffusion, DALL·E 3)

Diffusion models are currently **state-of-the-art in image generation**.

Idea

- They work by gradually adding noise to data (forward process).
- Then they train a model to reverse this process step by step — denoising until the image reappears.

Process

1. Start with a real image → add Gaussian noise in many steps until it's pure noise.
2. Train a neural network to predict and remove noise.
3. At generation time, start with noise and run the reverse process → get a clean synthetic image.

Why They Work Well

- High-quality, diverse images.
- Stable training compared to GANs.
- Used in **Stable Diffusion, MidJourney, DALL·E**.

Example: Give text prompt “A futuristic city at sunset” → diffusion model gradually denoises random pixels until it forms that cityscape.

Summary of Core Models:

- **Autoregressive (GPT):** Great for text.
- **VAEs:** Good for latent space + smooth generation.
- **GANs:** High-quality but hard to train.
- **Diffusion Models:** Current leader for images.

3. Applications of GenAI (8 minutes)

Text (LLMs)

- Chatbots, summarization, translation, coding assistants.

Images (GANs, Diffusion)

- Image synthesis, style transfer, design mockups.

Audio & Music (WaveNet, Jukebox, Suno)

- Text-to-speech, music generation.

Healthcare

- Drug molecule design via generative chemistry.
- Synthetic patient data for privacy-preserving ML.

Education & Business

- Personalized learning, marketing automation, content creation.

(If doing live, show a quick demo: e.g., generate text with ChatGPT, show an AI-generated image, or code snippet from Copilot.)

4. Benefits and Opportunities (5 minutes)

1. **Creativity Amplified** – AI as a co-creator.
2. **Productivity Boost** – automating repetitive tasks.
3. **Democratization** – lowering barriers for design, coding, content.
4. **Innovation Driver** – new drug designs, new media formats, personalized solutions.

---

5. Challenges and Risks (12 minutes)

5.1 Bias in Models

- Models reflect biases in training data.
- Example: word embeddings associating "doctor" with "male."

5.2 Misinformation & Deepfakes

- Generating fake content that looks real.
- Political, security, and societal risks.

5.3 Copyright & Ownership

- Training data scraped from the web raises legal disputes.
- Open question: who owns AI outputs?

5.4 Computational Cost & Environment

- Training LLMs like GPT-4 costs millions of dollars and huge carbon footprint.

5.5 Job Disruption

- Routine writing, design, and coding tasks may be automated.
- But creates new jobs: prompt engineering, AI safety, model auditing.

---

6. The Future of GenAI (5 minutes)

- **Multimodal Models**: like GPT-4V, Gemini, or Claude that process text + images + audio together.
- **AI Agents**: capable of reasoning, planning, and taking autonomous actions.
- **Edge AI**: running generative models on devices, not just cloud.
- **Regulation & Governance**: EU AI Act, U.S. AI executive orders.
- **Long-term research**: controllability, interpretability, alignment with human values.

Key takeaways:

- Generative AI = models that learn data distributions and generate new samples.
- Core techniques = Transformers, VAEs, GANs, Diffusion.
- Applications span text, images, audio, healthcare, education, and beyond.
- Challenges = bias, deepfakes, copyright, compute, ethics.
- Future = multimodal, agent-based, human-AI collaboration.

Final line:

“Generative AI is not just about machines replacing creativity. It’s about machines expanding what humans can imagine.”

Rizvi X – A Creative Initiative by Danial Rizvi