### 🎯 Objective:

Understand and experiment with the foundational techniques that power generative AI — including transformers, diffusion models, prompt design, multimodal architectures, and their evaluation.

## 🔍 Curriculum Breakdown

### 1. What is Generative AI?

#### 📌 1.1 Core Definition

* Generative AI refers to models that can generate original content such as:
  + Text (e.g., articles, poems)
  + Images (e.g., AI art, photorealistic images)
  + Code (e.g., scripts, functions, APIs)
  + Audio/Music (e.g., background scores, voices)
  + Video (e.g., scene generation, motion tracking)

#### 📌 1.2 Key Differences

* Generative models output novel data, not just classifications or predictions
* Often based on probabilistic learning, latent space exploration, and learned representations

🧠 *Mini-Reflection*: Contrast Linear Regression vs. GPT. What’s being predicted?

### 2. Transformer Architectures

#### 📌 2.1 Foundation Models

* GPT (decoder-only): autoregressive, next-token prediction
* BERT (encoder-only): masked token prediction for contextual embeddings
* T5 (encoder-decoder): sequence-to-sequence multitask learning

#### 📌 2.2 Attention Mechanism

* Self-attention: each token looks at all other tokens
* Positional Encoding: adds order information
* Scaled Dot-Product Attention and Multi-head attention

#### 📌 2.3 Use-Cases

* Summarization, translation, QA, chat interfaces
* Few-shot learning and transfer across tasks

📌 Lab: Use Hugging Face Transformers to try GPT-2 and T5 on summarization and question answering

### 3. Diffusion Models

#### 📌 3.1 Conceptual Workflow

* Forward process: add noise to data (e.g., images)
* Reverse process: learn how to denoise step-by-step
* Trained to reconstruct data from pure noise — effectively generate from scratch

#### 📌 3.2 Notable Models

* Stable Diffusion
* DALL·E, MidJourney
* Open-source pipelines via Hugging Face Diffusers library

#### 📌 3.3 Applications

* Text-to-image generation
* Inpainting: filling in masked image regions
* Style transfer and enhancement

📌 Exercise: Generate 5 unique images using Stable Diffusion + custom prompts

### 4. Prompt Engineering

#### 📌 4.1 Prompting Styles

* Zero-shot: "What is the capital of France?"
* Few-shot: Provide examples for structure
* Chain-of-Thought: encourage reasoning step-by-step

#### 📌 4.2 Prompt Design Tips

* Be specific about format, tone, persona
* Use delimiters, examples, and system messages

#### 📌 4.3 Tools for Prompting

* OpenAI Playground
* Hugging Face Inference API
* Prompt IDEs and testing platforms (PromptLayer, FlowGPT)

📌 Lab: Design three prompts to achieve the same goal with different styles (reasoning, creativity, structure)

### 5. Multimodal AI

#### 📌 5.1 What is Multimodal Learning?

* Models that understand and generate across multiple data types (text, image, audio)

#### 📌 5.2 Notable Multimodal Models

* CLIP: connects text and images via embeddings
* BLIP, BLIP-2: vision-language transformers
* GPT-4V (Vision): image + language prompts
* Flamingo, Gemini: text, image, video, interleaved understanding

#### 📌 5.3 Use-Cases

* Visual question answering
* Document parsing and captioning
* Image-grounded conversational agents

📌 Demo: Use Hugging Face Spaces for image captioning with BLIP-2

### 6. Evaluation Metrics in GenAI

#### 📌 6.1 Text Metrics

* BLEU: overlap in n-grams (machine translation)
* ROUGE: recall-based overlap for summarization
* METEOR: synonym-aware evaluation

#### 📌 6.2 Image Metrics

* FID (Fréchet Inception Distance): visual realism and diversity
* Inception Score (IS): quality and uniqueness

#### 📌 6.3 Human Evaluation

* Coherence, fluency, creativity
* Alignment with instructions and safety

📌 Activity: Evaluate multiple GenAI outputs using BLEU + human review side-by-side

### 7. Model Capabilities Showcase

| Task | Model Examples |
| --- | --- |
| Text Generation | ChatGPT, Claude, Mistral |
| Code Generation | Codex, GitHub Copilot |
| Image Generation | DALL·E 3, MidJourney, SDXL |
| Music/Audio | MusicLM, AudioCraft, Riffusion |
| Video (emerging) | Runway, Pika, Sora (OpenAI) |

📌 Challenge: Use 3 different GenAI APIs or models (text, image, code) in one mini-project

### 🧰 Tools & Libraries

* Hugging Face Transformers, Diffusers
* OpenAI APIs (ChatGPT, DALL·E)
* Replicate, Stability AI, DreamStudio
* PromptLayer, LangChain (prompt chaining)
* Colab + Gradio for quick interfaces

### 🧪 Capstone Mini-Project

Project: *Build a Prompt-Driven Generative App*

* Choose a domain: text, code, image
* Design creative prompts + use 1–2 APIs or models
* Build a simple front-end using Gradio or Streamlit
* Include evaluation (automated or human) and output examples
* Push to GitHub + share on Hugging Face Spaces (optional)