Key Models in Generative AI: GPT, DALL·E, Codex, Stable Diffusion, GANs and Transformers

1. Introduction to Generative AI

1.1 Definition and Scope

Generative AI refers to a class of artificial intelligence systems capable of creating new content—such as text, images, music, code, and video—by learning patterns from existing data. These models don't just analyze or classify; they synthesize novel outputs that mimic human creativity.

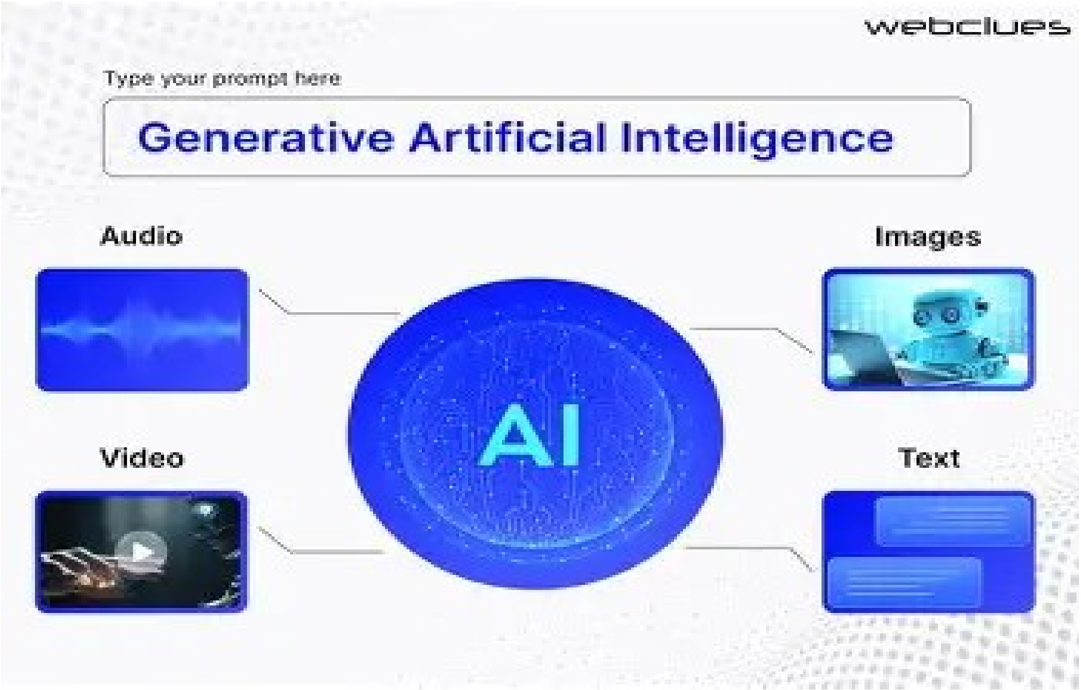
1.2 Evolution of Generative Models

Early AI models focused on rule-based systems and statistical learning. The advent of deep learning, particularly neural networks, enabled the development of generative models like Variational Autoencoders (VAEs), GANs, and Transformers. These models can now generate content that is often indistinguishable from human-created work.

1.3 Importance and Applications

Generative AI is transforming industries:

- **Media & Entertainment**: Scriptwriting, music composition, and video generation.
- **Design & Art**: Logo creation, digital painting, and fashion design.
- **Software Development**: Code generation and debugging.
- **Education**: Personalized learning materials and tutoring.



2. GPT (Generative Pre-trained Transformer)

2.1 What Is GPT?

GPT is a family of autoregressive language models developed by OpenAI. It generates human-like text by predicting the next word in a sequence based on the context of previous words.

2.2 Architecture

GPT is built on the Transformer architecture, which uses self-attention mechanisms to model relationships between words in a sequence. This allows it to understand context and maintain coherence over long passages.

2.3 Training Methodology

GPT undergoes two main phases:

- **Pre-training**: The model is trained on a large corpus of text data to learn general language patterns.
- **Fine-tuning**: It is then adapted to specific tasks (e.g., summarization, translation) using smaller, task-specific datasets.

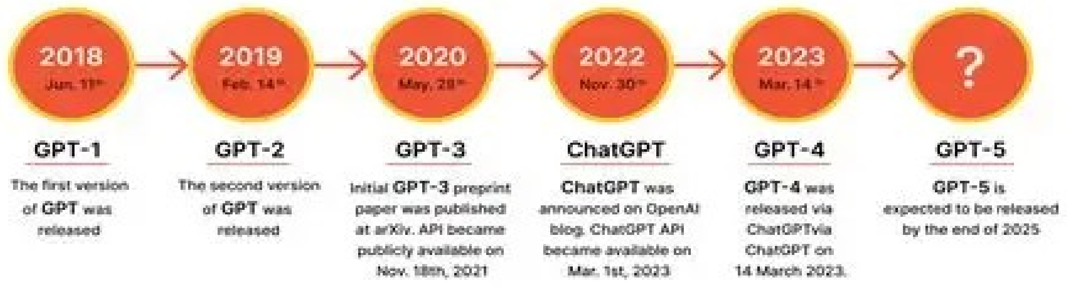
2.4 Capabilities

GPT can:

- Generate essays, stories, and articles.
- Answer questions and summarize documents.
- Translate languages and complete sentences.
- Engage in coherent conversations.

2.5 Real-World Applications

- **Chatbots and Virtual Assistants**: Powering tools like ChatGPT.
- **Content Creation**: Assisting writers and marketers.
- **Education**: Providing explanations and tutoring.
- **Business Automation**: Drafting emails, reports, and documentation.


2018
Jun. 11th
2019
Feb. 14th
2020
May. 28th
2022
Nov. 30th
2023
Mar. 14th
?
GPT-1
The first version of GPT was released
GPT-2
The second version of GPT was released
GPT-3
Initial GPT-3 preprint paper was published at arXiv. API became publicly available on Nov. 18th, 2021
ChatGPT
ChatGPT was announced on OpenAI blog. ChatGPT API became available on Mar. 1st, 2023
GPT-4
GPT-4 was released via ChatGPTvia ChatGPT on 14 March 2023.
GPT-5
GPT-5 is expected to be released by the end of 2025

3. DALL·E

3.1 What Is DALL·E?

DALL·E is a multimodal generative model developed by OpenAI that creates images from textual descriptions. It enables users to visualize concepts by simply describing them in natural language.

3.2 Architecture

DALL·E combines:

- **Transformer-based text encoder**: Understands the semantics of the input prompt.
- **VQ-VAE (Vector Quantized Variational Autoencoder)**: Decodes the text representation into an image.

3.3 CLIP Integration

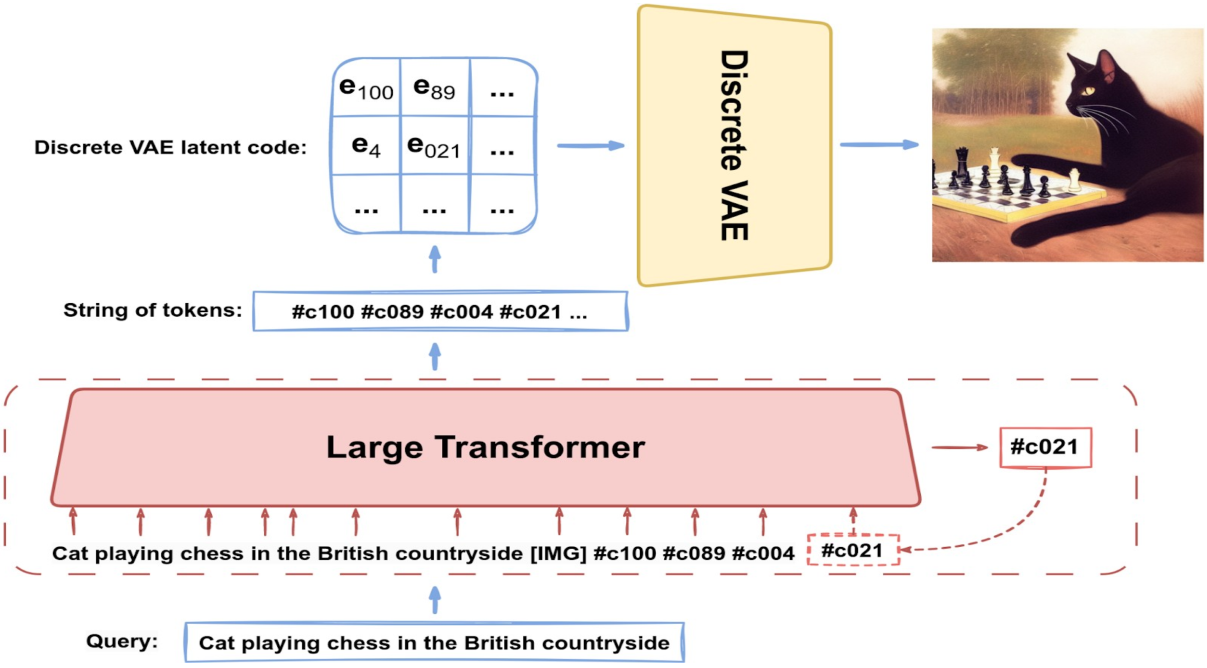
DALL·E uses CLIP (Contrastive Language–Image Pre-training) to align text and image embeddings. This ensures that the generated image accurately reflects the meaning of the prompt.

3.4 Capabilities

- Generates novel and imaginative images.
- Performs inpainting (editing parts of an image).
- Creates variations of existing images based on prompts.

3.5 Applications

- **Advertising and Marketing**: Visualizing campaign ideas.
- **Education**: Creating illustrations for learning materials.
- **Art and Design**: Assisting artists in concept development.
- **Product Development**: Prototyping and ideation.


Discrete VAE latent code:
e_{100} e_{89} ...
e_4 e_{021} ...
... ...
Discrete VAE
String of tokens:
#c100 #c089 #c004 #c021 ...
Large Transformer
#c021
Cat playing chess in the British countryside [IMG] #c100 #c089 #c004
#c021
Query:
Cat playing chess in the British countryside

4. Codex

4.1 What Is Codex?

Codex is a language model fine-tuned from GPT-3, specifically trained on code. It translates natural language instructions into executable code, bridging the gap between human language and programming.

4.2 Architecture and Training

Codex uses the Transformer architecture and is trained on billions of lines of code from public repositories like GitHub. It learns programming syntax, logic, and common patterns across multiple languages.

4.3 Capabilities

- Writes code from plain English prompts.
- Completes and refactors code.
- Explains code functionality.
- Supports multiple languages including Python, JavaScript, and Go.

4.4 Applications

- **GitHub Copilot**: Assists developers by suggesting code in real-time.
- **Education**: Helps students learn programming interactively.
- **Automation**: Generates scripts for repetitive tasks.
- **Prototyping**: Quickly builds functional code for testing ideas.

5. Stable Diffusion

5.1 What Is Stable Diffusion?

Stable Diffusion is an open-source text-to-image model developed by Stability AI. It generates high-quality images from textual prompts using a diffusion-based approach.

5.2 Diffusion Process

The model starts with random noise and refines it step-by-step into a coherent image. This reverse diffusion process is guided by a text prompt and learned during training.

5.3 Latent Space Optimization

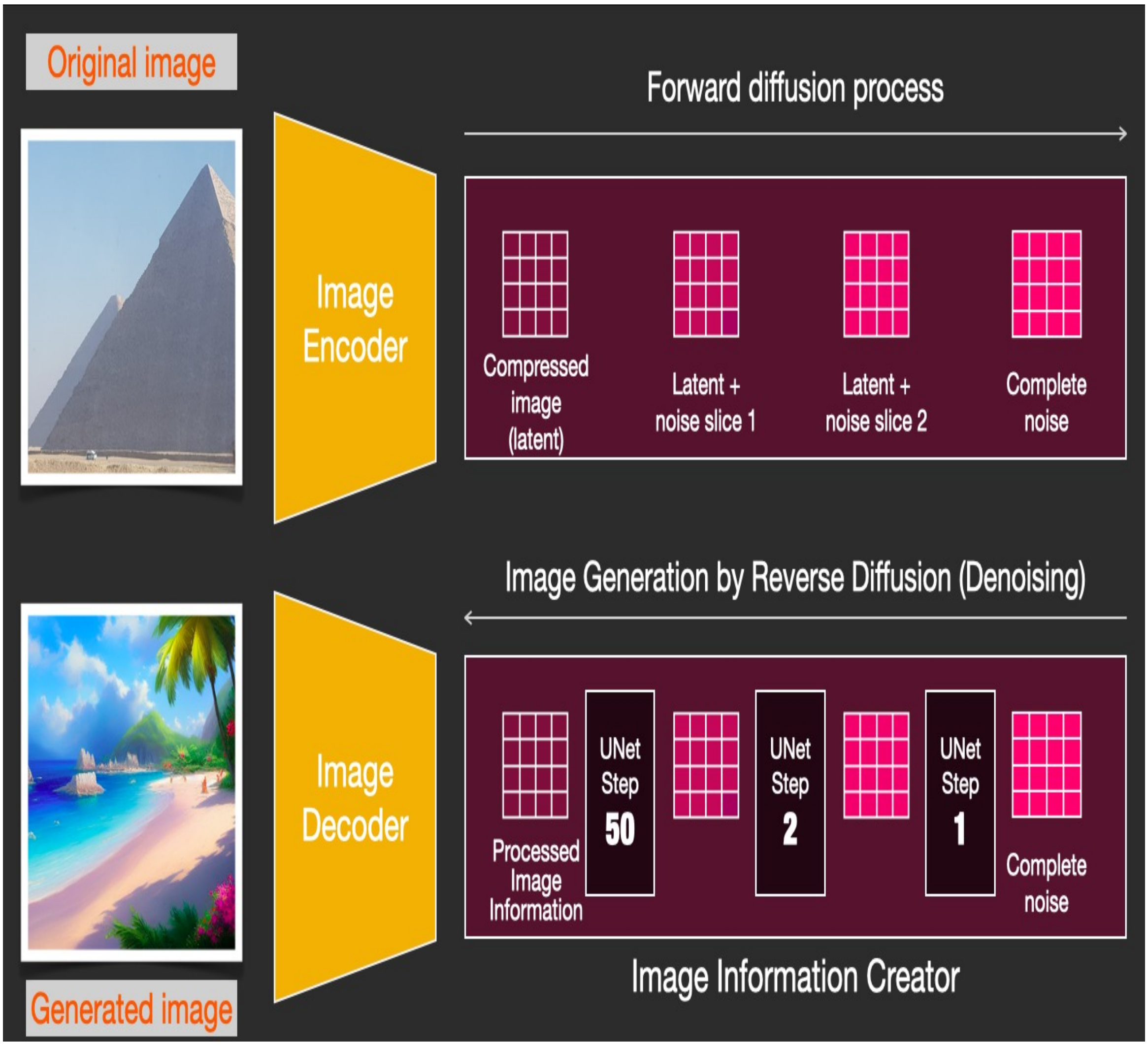
Unlike pixel-based models, Stable Diffusion operates in a compressed latent space. This reduces computational load and allows the model to run on consumer-grade hardware.

5.4 Architecture

- **U-Net**: Performs the denoising steps.
- **VAE**: Encodes and decodes images.
- **CLIP**: Guides image generation based on text.

5.5 Applications

- **Digital Art**: Creating illustrations and concept art.
- **Game Design**: Generating assets and environments.
- **Marketing**: Producing visuals for campaigns.
- **Customization**: Fine-tuning for specific styles or themes.



6. Diffusion Models

6.1 Concept

Diffusion models generate data by learning to reverse a noise process. They start with random noise and apply a series of denoising steps to produce a structured output.

6.2 Training Process

The model is trained to predict the noise added at each step of the forward process. During inference, it reverses this process to generate new data.

6.3 Advantages

- **High Fidelity**: Produces detailed and realistic images.
- **Stability**: More stable training compared to GANs.
- **Control**: Easier to guide generation using conditioning (e.g., text prompts).

6.4 Applications

- **Image Generation**: Used in models like Stable Diffusion and Imagen.
- **Audio Synthesis**: Generating speech and music.
- **Scientific Modeling**: Simulating molecular structures and physical systems.

7. GANs (Generative Adversarial Networks)

7.1 What Are GANs?

GANs are generative models that consist of two neural networks—a generator and a discriminator—that compete in a zero-sum game to produce realistic data.

7.2 Architecture

- **Generator**: Creates synthetic data from random noise.
- **Discriminator**: Evaluates whether the data is real or fake.

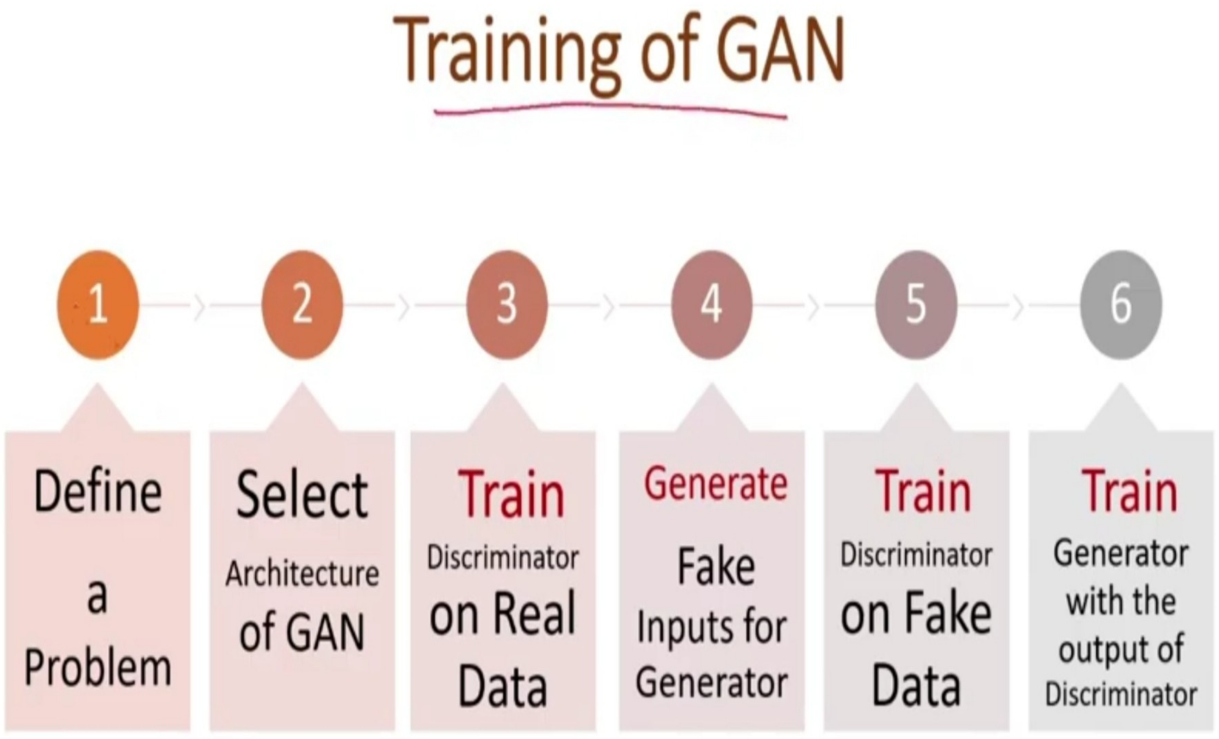
The generator improves by learning to fool the discriminator, while the discriminator improves by learning to detect fakes.

7.3 Variants

- **DCGAN**: Uses convolutional layers for image generation.
- **StyleGAN**: Enables fine-grained control over image features.
- **CycleGAN**: Translates images between domains (e.g., summer to winter scenes).

7.4 Applications

- **Deepfakes**: Creating realistic face swaps.
- **Art and Design**: Generating stylized images.
- **Data Augmentation**: Enhancing training datasets.
- **Super-Resolution**: Upscaling low-resolution images.


Training of GAN
1
2
3
4
5
6
Define a Problem
Select Architecture of GAN
Train Discriminator on Real Data
Generate Fake Inputs for Generator
Train Discriminator on Fake Data
Train Generator with the output of Discriminator

8. Transformers

8.1 Introduction

Transformers are a neural network architecture introduced in 2017 that revolutionized sequence modeling. They are the foundation of models like GPT, Codex, and DALL·E.

8.2 Self-Attention Mechanism

Self-attention allows the model to weigh the importance of each token in a sequence relative to others. This enables it to capture long-range dependencies and contextual relationships.

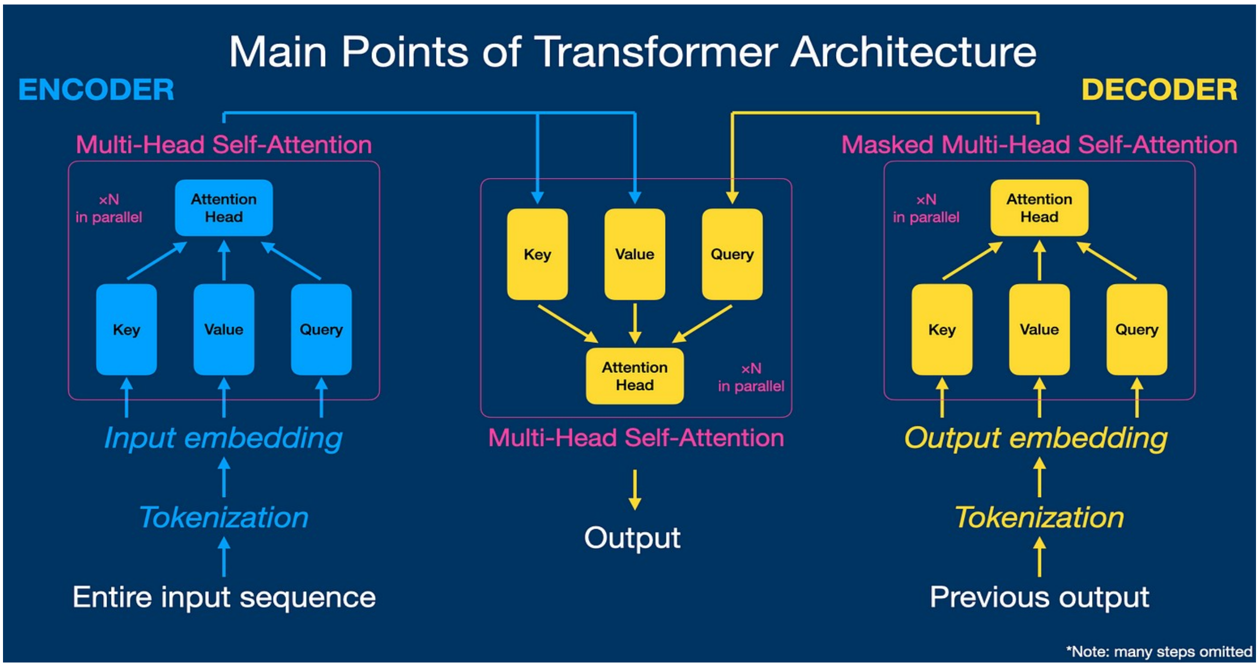
8.3 Architecture Components

- **Encoder**: Processes input sequences.
- **Decoder**: Generates output sequences.
- **Multi-head Attention**: Focuses on different parts of the input simultaneously.
- **Feed-forward Layers**: Apply transformations to the attention outputs.
- **Positional Encoding**: Adds information about token order.

8.4 Applications

- **Natural Language Processing**: Translation, summarization, question answering.
- **Computer Vision**: Vision Transformers (ViTs) for image classification.
- **Multimodal AI**: Combining text, image, and audio inputs.

- **Speech and Audio**: Transcription and synthesis.


Main Points of Transformer Architecture
ENCODER
DECODER
Multi-Head Self-Attention
×N
in parallel
Attention Head
Key
Value
Query
Input embedding
Tokenization
Entire input sequence
Key
Value
Query
Attention Head
×N
in parallel
Multi-Head Self-Attention
Output
Masked Multi-Head Self-Attention
×N
in parallel
Attention Head
Key
Value
Query
Output embedding
Tokenization
Previous output
*Note: many steps omitted