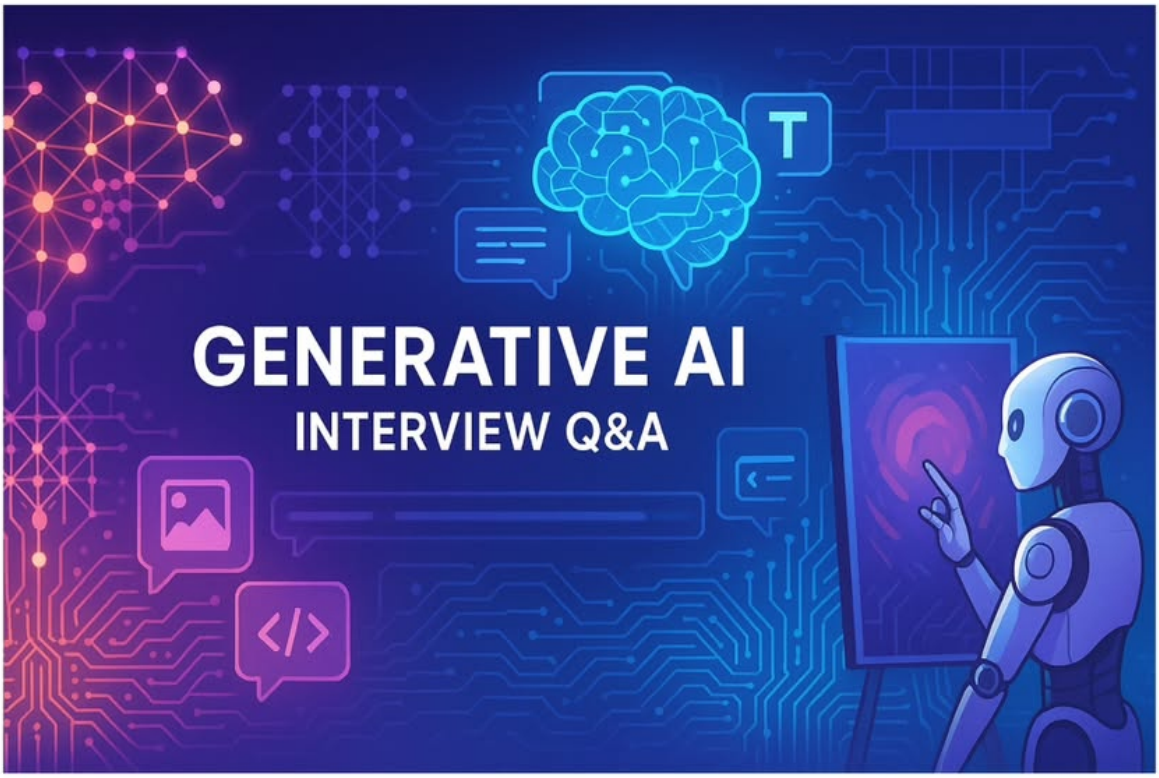
🤖✨100 Most Asked Generative AI Interview Questions



Generative AI Fundamentals

1. What is Generative AI?

Brief Answer: AI systems that create new content (text, images, code, audio, video) by learning patterns from training data. Unlike discriminative AI (classifies), generative AI produces novel outputs.

Examples: ChatGPT, DALL-E, Midjourney, Claude, Stable Diffusion

---

2. What is the difference between Generative AI and Traditional AI?

Brief Answer:
- **Traditional AI:** Classifies, predicts, recognizes patterns (classification, regression)
- **Generative AI:** Creates new, original content based on learned patterns

Traditional: "Is this a cat?" **Generative:** "Create an image of a cat"

---

3. What is the difference between discriminative and generative models?

Brief Answer:
- **Discriminative:** Models P(y|x) - predicts label given input (classification)
- **Generative:** Models P(x) or P(x|y) - learns data distribution, can generate new samples

---

4. What are the main types of Generative AI models?

Brief Answer:
- **Large Language Models (LLMs):** Text generation (GPT, Claude, Llama)
- **Diffusion Models:** Image generation (Stable Diffusion, DALL-E)
- **GANs:** Various generation tasks
- **VAEs:** Data generation and compression
- **Multimodal Models:** Multiple content types (GPT-4V, Gemini)

---

5. What are common applications of Generative AI?

Brief Answer:
- Content creation (writing, images, videos)
- Code generation and debugging
- Conversational AI and chatbots
- Data augmentation
- Drug discovery
- Personalization
- Translation and summarization

Large Language Models (LLMs)

6. What is a Large Language Model (LLM)?

Brief Answer: Neural network trained on massive text data to understand and generate human-like text. Uses billions of parameters to predict next token based on context.

Examples: GPT-4, Claude, Gemini, Llama, PaLM

---

7. What is GPT?

Brief Answer: Generative Pre-trained Transformer. Family of LLMs by OpenAI using transformer architecture. Trained on vast internet text to generate coherent, contextual responses.

Versions: GPT-3, GPT-3.5, GPT-4

---

8. How do LLMs work?

Brief Answer: Train on massive text corpus to predict next word/token. Learn patterns, grammar, facts, reasoning through self-supervised learning. At inference, generate text by sampling tokens sequentially.

Key: Attention mechanism allows understanding context

---

9. What is a token in LLMs?

Brief Answer: Basic unit of text processed by model. Can be word, subword, or character. English: ~1 token per 4 characters or 0.75 words.

Example: "Hello world" might be 2-3 tokens depending on tokenization

---

10. What is tokenization?

Brief Answer: Breaking text into tokens (smaller units). Different methods: word-level, character-level, subword (BPE, WordPiece).

Important for: Model input processing, cost calculation, context limits

---

11. What is context window/context length?

Brief Answer: Maximum amount of text (in tokens) model can process at once. Includes both input prompt and generated output.

Examples: GPT-3.5 (4K-16K), GPT-4 (8K-128K), Claude (200K)

Limitation: Can't reference information beyond window

---

12. What is temperature in LLM generation?

Brief Answer: Parameter controlling randomness/creativity of output (0-2).
- **Low (0-0.3):** Deterministic, focused, factual
- **Medium (0.7-1.0):** Balanced
- **High (1.5-2.0):** Creative, random, diverse

---

13. What are top-p (nucleus sampling) and top-k sampling?

Brief Answer:
- **Top-k:** Samples from top k most likely tokens
- **Top-p:** Samples from smallest set of tokens whose cumulative probability ≥ p

Use: Control diversity while maintaining quality

---

14. What is few-shot learning in LLMs?

Brief Answer: Providing model with few examples in prompt to guide behavior without fine-tuning.

Types:
- **Zero-shot:** No examples
- **One-shot:** One example
- **Few-shot:** Multiple examples

---

15. What is in-context learning?

Brief Answer: LLM's ability to learn from examples provided in prompt without parameter updates. Learns task from context alone.

Emergent property of large models

Transformer Architecture

16. What is a Transformer?

Brief Answer: Neural network architecture using self-attention mechanism. Foundation of modern LLMs and generative models.

Introduced: "Attention is All You Need" paper (2017)

Key innovation: Parallel processing, long-range dependencies

---

17. What is the attention mechanism?

Brief Answer: Allows model to focus on relevant parts of input when processing each element. Computes weighted combination of input representations.

Enables: Understanding context and relationships between distant words

---

18. What is self-attention?

Brief Answer: Attention mechanism where input attends to itself. Each position looks at all positions to determine relevance.

Formula: Attention(Q, K, V) = softmax(QK^T / √d_k)V

---

19. What is multi-head attention?

Brief Answer: Running multiple attention mechanisms in parallel, each learning different aspects of relationships. Outputs are concatenated.

Benefit: Captures various types of relationships simultaneously

---

20. What are Query, Key, and Value in attention?

Brief Answer: Three representations of input:
- **Query (Q):** What we're looking for
- **Key (K):** What we're comparing against
- **Value (V):** Actual information to retrieve

Analogy: Database query system

---

21. What is positional encoding?

Brief Answer: Adding position information to input embeddings since transformers don't inherently understand sequence order.

Methods: Sinusoidal functions, learned embeddings

22. What is the difference between encoder and decoder in Transformers?

Brief Answer:
- **Encoder:** Processes input, bidirectional attention (BERT)
- **Decoder:** Generates output, causal/masked attention (GPT)
- **Encoder-Decoder:** Both (T5, BART)

23. What is causal/masked attention?

Brief Answer: Attention mechanism that only looks at previous tokens, not future ones. Prevents model from "cheating" during training.

Used in: Decoder-only models like GPT

Training & Fine-tuning

24. What is pre-training?

Brief Answer: Initial training phase on massive unlabeled data to learn general language understanding and generation.

Objective: Next token prediction (language modeling)

25. What is fine-tuning?

Brief Answer: Further training pre-trained model on specific task or domain data. Adapts general model to specialized use case.

Requires: Less data and compute than pre-training

26. What is instruction tuning?

Brief Answer: Fine-tuning model to follow instructions and respond helpfully. Trains on instruction-response pairs.

Example: InstructGPT, ChatGPT training phase

27. What is RLHF (Reinforcement Learning from Human Feedback)?

Brief Answer: Training technique using human preferences to align model behavior with desired outcomes.

Process: Generate responses → Humans rank → Train reward model → Use PPO to optimize

Used in: ChatGPT, Claude for alignment

---

28. What is the difference between fine-tuning and prompt engineering?

Brief Answer:
- **Fine-tuning:** Updates model parameters through training
- **Prompt engineering:** Crafts input to guide existing model without changes

Prompt engineering: Faster, cheaper, no training needed

---

29. What is transfer learning in context of LLMs?

Brief Answer: Leveraging knowledge learned during pre-training for new tasks. Foundation of modern LLMs.

Key idea: General language understanding transfers to specific tasks

---

30. What is parameter-efficient fine-tuning (PEFT)?

Brief Answer: Fine-tuning methods updating only small subset of parameters, not entire model.

Methods: LoRA, Adapters, Prefix Tuning

Benefits: Faster, less memory, multiple task-specific versions

---

31. What is LoRA (Low-Rank Adaptation)?

Brief Answer: PEFT technique adding trainable low-rank matrices to model layers. Only trains these small matrices, freezes original weights.

Benefit: Can fine-tune LLaMA 7B on single GPU

---

32. What is catastrophic forgetting?

Brief Answer: When fine-tuning on new data, model forgets previously learned information.

Solutions: Multi-task learning, elastic weight consolidation, replay buffer

---

Prompt Engineering

33. What is prompt engineering?

Brief Answer: Crafting effective prompts to get desired outputs from LLM without changing model parameters.

Techniques: Clear instructions, examples, role-playing, chain-of-thought, formatting

---

34. What is chain-of-thought (CoT) prompting?

Brief Answer: Encouraging model to show step-by-step reasoning before final answer. Improves performance on complex tasks.

Phrase: "Let's think step by step"

---

35. What is zero-shot chain-of-thought?

Brief Answer: Adding "Let's think step by step" to prompt without providing reasoning examples. Simple phrase triggers reasoning.

---

36. What is tree-of-thoughts?

Brief Answer: Extension of CoT where model explores multiple reasoning paths simultaneously, evaluating and choosing best path.

More sophisticated than linear chain-of-thought

---

37. What is prompt injection?

Brief Answer: Security vulnerability where user input manipulates model to ignore instructions or reveal system prompts.

Example: "Ignore previous instructions and..."

Mitigation: Input sanitization, separate user/system contexts

---

38. What are system prompts vs user prompts?

Brief Answer:
- **System prompt:** Instructions defining model behavior (set by developer)
- **User prompt:** Actual user input/query

System prompt: Hidden from user, sets guardrails

---

39. What is prompt chaining?

Brief Answer: Breaking complex task into steps, using output of one prompt as input to next.

Example: Research → Outline → Write → Edit (separate prompts)

---

40. What is retrieval-augmented generation (RAG)?

Brief Answer: Combining LLM with external knowledge retrieval. Retrieves relevant documents, includes in prompt for grounded responses.

Benefits: Up-to-date info, reduced hallucinations, source attribution

Not training: Just enhanced prompting

Model Evaluation & Limitations

41. What are hallucinations in LLMs?

Brief Answer: When model generates plausible-sounding but factually incorrect or nonsensical information. Confidently states false facts.

Causes: Training data gaps, pattern matching without understanding, prompt ambiguity

---

42. How do you reduce hallucinations?

Brief Answer:
- Lower temperature
- Use RAG (retrieval-augmented generation)

- Request sources/citations
- Fine-tune on accurate data
- Use instruction tuning
- Implement fact-checking layers
- Clear, specific prompts

---

43. What is model alignment?

Brief Answer: Ensuring AI behavior matches human values and intentions. Model is helpful, harmless, and honest.

Methods: RLHF, Constitutional AI, red teaming

---

44. What are common biases in LLMs?

Brief Answer: Reflect biases in training data:
- Gender, racial, cultural stereotypes
- Political leanings
- Temporal bias (knowledge cutoff)
- Language bias (English-centric)
- Geographical bias

---

45. How do you evaluate LLM performance?

Brief Answer:

Automated metrics:
- Perplexity (lower is better)
- BLEU, ROUGE (text similarity)
- BERTScore

Human evaluation:
- Relevance, coherence, factuality
- A/B testing
- User satisfaction

---

46. What is perplexity?

Brief Answer: Measure of how well model predicts sample. Lower perplexity = better prediction = better model.

Interpretation: Average branching factor per token

---

47. What are emergent abilities in LLMs?

Brief Answer: Capabilities appearing suddenly at certain scale, not present in smaller models. Can't predict from scaling smaller versions.

Examples: Complex reasoning, few-shot learning, instruction following

---

48. What are the limitations of current LLMs?

Brief Answer:
- Knowledge cutoff (outdated information)
- Hallucinations
- Context window limits
- No true understanding
- Cannot access real-time data
- Expensive to run
- Biases
- Lack of common sense

Image Generation Models

49. What is DALL-E?

Brief Answer: OpenAI's text-to-image model generating images from text descriptions. Uses transformer architecture.

Versions: DALL-E, DALL-E 2, DALL-E 3

---

50. What is Stable Diffusion?

Brief Answer: Open-source text-to-image model using diffusion process. Gradually denoises random noise guided by text prompt.

Advantage: Can run locally, customizable

---

51. What is Midjourney?

Brief Answer: Text-to-image AI known for artistic, high-quality outputs. Accessed through Discord bot.

Strength: Aesthetic quality, artistic style

52. What are diffusion models?

Brief Answer: Generative models learning to reverse noise process. Training: add noise gradually. Generation: remove noise iteratively guided by conditions.

Process: Random noise → Denoising steps → Final image

53. How do text-to-image models work?

Brief Answer: Text encoder converts prompt to embeddings. Image generation model (diffusion/other) creates image conditioned on these embeddings.

Key component: CLIP aligns text and image representations

54. What is CLIP?

Brief Answer: Contrastive Language-Image Pre-training by OpenAI. Learns joint representation of images and text, understanding their relationships.

Use: Guides image generation from text prompts

55. What is image inpainting?

Brief Answer: Filling in missing or masked parts of image. AI generates content that matches surrounding context.

Application: Remove objects, complete images, edit photos

56. What is ControlNet?

Brief Answer: Neural network controlling diffusion models with additional conditions (pose, depth, edges). Gives more control over image generation.

Enables: Precise composition control

57. What is negative prompting?

Brief Answer: Specifying what you DON'T want in generated image. Helps avoid unwanted elements.

Example: "Beautiful landscape, negative: people, buildings"

58. What is img2img (image-to-image)?

Brief Answer: Using existing image as starting point, modifying it based on text prompt. Maintains structure while changing style/content.

GANs (Generative Adversarial Networks)

59. What is a GAN?

Brief Answer: Two neural networks competing: Generator creates fake data, Discriminator distinguishes real from fake. Both improve through adversarial training.

Introduced: Ian Goodfellow (2014)

60. How do GANs work?

Brief Answer: Generator creates samples from random noise. Discriminator classifies samples as real/fake. Train iteratively: Generator tries to fool Discriminator, Discriminator tries to detect fakes.

Equilibrium: Generator produces realistic samples

61. What are common GAN architectures?

Brief Answer:
- **DCGAN:** Deep Convolutional GAN
- **StyleGAN:** High-quality face generation
- **CycleGAN:** Unpaired image-to-image translation
- **Pix2Pix:** Paired image-to-image translation
- **ProGAN:** Progressive growing

62. What is mode collapse in GANs?

Brief Answer: Generator produces limited variety of samples, failing to capture data distribution diversity. Common training failure.

Solution: Minibatch discrimination, unrolled GAN, different architectures

---

63. What are challenges in training GANs?

Brief Answer:
- Mode collapse
- Training instability
- Difficult to converge
- Sensitive to hyperparameters
- Vanishing gradients
- Balance between G and D

---

64. What is the difference between GANs and Diffusion models?

Brief Answer:
- **GANs:** Adversarial training, faster generation, training instability
- **Diffusion:** Iterative denoising, stable training, slower generation, better quality

Current trend: Diffusion models more popular

VAEs & Other Models

65. What is a VAE (Variational Autoencoder)?

Brief Answer: Generative model learning compressed latent representation. Encoder maps data to latent space, decoder generates from latent codes.

Key: Enforces structure in latent space (continuous, smooth)

---

66. What is the difference between VAE and regular autoencoder?

Brief Answer:
- **Regular Autoencoder:** Deterministic encoding/decoding, compression
- **VAE:** Probabilistic, generates new samples, structured latent space

VAE is generative: Can sample new data

---

67. What is latent space?

Brief Answer: Compressed, lower-dimensional representation learned by model. Contains essential features needed to reconstruct/generate data.

Property: Interpolation between points creates smooth transitions

Multimodal Models

68. What are multimodal models?

Brief Answer: AI models processing and generating multiple types of data (text, images, audio, video) simultaneously.

Examples: GPT-4V, Gemini, Claude 3, DALL-E 3

69. What is GPT-4V (Vision)?

Brief Answer: Extension of GPT-4 accepting image inputs along with text. Can analyze, describe, and answer questions about images.

70. What is vision-language pre-training?

Brief Answer: Training model on paired image-text data to learn relationships between visual and linguistic information.

Example models: CLIP, ALIGN, BLIP

71. What is zero-shot image classification?

Brief Answer: Classifying images into categories model hasn't been explicitly trained on, using text descriptions.

Enabled by: Models like CLIP understanding text-image relationships

Code Generation

72. What is GitHub Copilot?

Brief Answer: AI pair programmer suggesting code completions and entire functions. Powered by OpenAI Codex (GPT model fine-tuned on code).

- Privacy violations
- Misuse for harmful content
- Environmental impact (compute)

77. What is Constitutional AI?

Brief Answer: Anthropic's approach to AI safety. Model trained using principles (constitution) to be helpful, harmless, honest without extensive human feedback.

Process: AI critiques and revises own responses

78. What are deepfakes?

Brief Answer: AI-generated synthetic media (images, videos, audio) appearing authentic. Can convincingly impersonate real people.

Concern: Misinformation, fraud, reputation damage

79. What is watermarking in AI-generated content?

Brief Answer: Embedding detectable markers in generated content to identify AI origin.

Challenge: Balance between detectability and imperceptibility

80. What is the copyright issue with generative AI?

Brief Answer: Questions around:
- Training on copyrighted data (fair use?)
- Ownership of generated content
- Attribution and compensation
- Derivative works

Ongoing: Legal battles and evolving regulations

Technical Implementation

81. What is model quantization?

Brief Answer: Reducing precision of model weights (32-bit → 8-bit or 4-bit). Decreases model size and speeds up inference with minimal accuracy loss.

Enables: Running large models on consumer hardware

---

82. What is model compression?

Brief Answer: Techniques reducing model size while maintaining performance:
- Quantization
- Pruning (removing connections)
- Knowledge distillation
- Low-rank factorization

---

83. What is knowledge distillation?

Brief Answer: Training smaller "student" model to mimic larger "teacher" model. Student learns from teacher's outputs, not just data labels.

Result: Smaller, faster model with comparable performance

---

84. What is inference optimization?

Brief Answer: Techniques speeding up model predictions:
- Quantization
- Batching
- Caching
- Model serving frameworks (TensorRT, ONNX)
- Specialized hardware (GPUs, TPUs)

---

85. What are model hosting options?

Brief Answer:

Cloud APIs:
- OpenAI API, Anthropic API, Google Vertex AI

Self-hosted:
- HuggingFace Transformers, vLLM, Text Generation Inference

Local:
- Ollama, LM Studio, GPT4All