

Index

- Study Guide
 - Introduction
 - Target candidate description
 - Recommended AWS knowledge
 - Exam content
 - Question types
 - Unscored content
 - Exam results
 - Content outline
 - Domain 1: Fundamentals of AI and ML
 - Explain basic AI concepts and terminologies
 - Identify practical use cases for AI
 - Describe the ML development lifecycle
 - Domain 2: Fundamentals of Generative AI
 - Explain the basic concepts of generative AI
 - Understand the capabilities and limitations of generative AI for solving business problems
 - Describe AWS infrastructure and technologies for building generative AI applications
 - Domain 3: Applications of Foundation Models
 - Describe design considerations for applications that use foundation models
 - Choose effective prompt engineering techniques
 - Describe the training and fine-tuning process for foundation models
 - Describe methods to evaluate foundation model performance
 - Domain 4: Guidelines for Responsible AI
 - Explain the development of AI systems that are responsible
 - Recognize the importance of transparent and explainable models
 - Domain 5: Security, Compliance, and Governance for AI Solutions
 - Explain methods to secure AI systems
 - Recognize governance and compliance regulations for AI systems
- Cloud Computing
 - What is Cloud Computing?
 - The Deployment Models of the Cloud
 - The Five Characteristics of Cloud Computing
 - Six Advantages of Cloud Computing
 - Problems Solved by the Cloud
 - Types of Cloud Computing
 - Example of Cloud Computing Types
 - Pricing of the Cloud – Quick Overview
 - How Cloud Pricing Solves Traditional IT Cost Issues
 - AWS Cloud Use Cases
- GenAI Introduction
 - What is Generative AI?
 - How it differs from traditional AI
 - Real-world applications
 - Why it's revolutionary
 - Foundation Model
 - Large Language Models
 - What makes them “large”
 - How to use them
 - Important characteristics

AWS Certified AI Practitioner (AIF-C01) Study Notes

- Popular LLM examples
- Generative Language Models
 - The way they work
 - Example of next-word prediction
 - Key characteristics
 - How they're trained
 - The power of generative models
- GenAI for Images
 - How diffusion models work
 - What they can do
- Amazon Bedrock
 - What is Amazon Bedrock?
 - Foundation Models
 - How to choose a foundation model?
 - Key Selection Factors
 - Important Considerations
 - Model-Specific Details
 - Amazon Titan
 - Llama-2 (Meta)
 - Claude (Anthropic)
 - Stable Diffusion (Stability AI)
 - Selection Strategy
 - Fine-Tuning a Model
 - Types of Fine-Tuning
 - 1. Instruction-Based Fine-Tuning
 - 2. Continued Pre-training
 - 3. Single-Turn and Multi-Turn Messaging
 - Transfer Learning
 - Use Cases for Fine-Tuning
 - Evaluating Foundation Models
 - Automatic Evaluation
 - Human Evaluation
 - What is RAG (Retrieval-Augmented Generation)?
 - How RAG Works with Amazon Bedrock
 - Benefits of RAG
 - What is a Vector Database?
 - Available Vector Database Options
 - Data Sources Supported by Amazon Bedrock
 - Vector Embeddings and Document Processing
 - Use Cases for Amazon Bedrock with RAG
 - RAG vs Fine-Tuning Comparison
 - When to Use RAG vs Fine-Tuning
 - Guardrails
 - Bedrock Agents
 - Bedrock - Other Features
 - Amazon Bedrock & CloudWatch
 - Pricing
 - Pricing Modes
 - Cost Impact of Model Improvement Approaches
 - Cost Optimization Strategies
- Prompt Engineering
 - What is Prompt Engineering?

GenAI Introduction

What is Generative AI?

- Generative AI is a branch of Deep Learning
- The idea is simple: you train a model on existing data, and it learns to create new data that looks similar
- Think of it like teaching someone to paint by showing them thousands of paintings
- Once trained, it can create brand new content that follows the same patterns

How it differs from traditional AI

- Traditional AI: Analyzes and classifies existing data (“Is this a cat or dog?”)
- Generative AI: Creates new data (“Generate a picture of a cat playing piano”)
- It’s like the difference between sorting mail and writing letters

Real-world applications

- Chatbots: Having conversations (ChatGPT, Claude)
- Code generation: Writing code from descriptions (GitHub Copilot)
- Image creation: Making art and photos (DALL-E, Midjourney)
- Music composition: Creating original music
- Content writing: Blogs, emails, marketing copy
- Video generation: Creating video clips from text

Why it’s revolutionary

- Can automate creative tasks that humans do
- Works across multiple types of content (text, images, audio, video)
- Gets better over time as models improve
- Makes advanced AI accessible to everyone through simple prompts
- Can combine ideas in ways humans might not think of
- You can train these models on almost anything:
 - Text: Books, articles, conversations
 - Images: Photos, paintings, illustrations
 - Audio: Music, speech, sound effects
 - Code: Programming languages and scripts
 - Video: Movies, TV shows, clips
 - And more: 3D models, scientific data, etc.

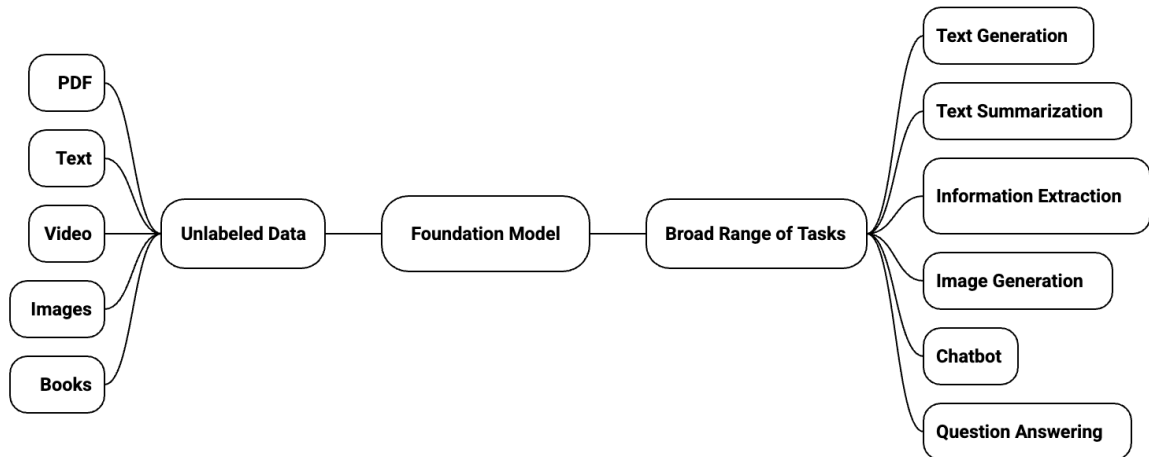
Foundation Model

- To create anything, you need a Foundation Model
- These models are trained on huge amounts of data from different sources
- Training one is expensive - think tens of millions of dollars
- Examples: GPT-4o is the foundation model that powers ChatGPT
- You can find foundation models from big companies like:
 - OpenAI
 - Meta (Facebook)
 - Amazon
 - Google

AWS Certified AI Practitioner (AIF-C01) Study Notes

- Anthropic
- And many others
- Some are free and open (like Meta's Llama, Google's BERT), while others cost money to use (like OpenAI's GPT-4o)

Foundation Model Training and Applications



Large Language Models

- These are AI models that generate text that sounds like a human wrote it (like ChatGPT)
- They learn from massive amounts of text - books, articles, websites, basically everything
- They're huge models with billions of parameters

What makes them “large”

- GPT-3 has 175 billion parameters
- GPT-4 has even more (exact number is not publicly disclosed)
- Parameters are like the knobs that the model adjusts during training
- More parameters usually means better understanding and more capabilities
- But also means more computing power and cost to run
- They can do all sorts of language tasks:
 - Translation: Convert text from one language to another
 - Summarization: Condense long articles into short summaries
 - Answering questions: Respond to questions based on their training
 - Creating content: Write stories, emails, code, etc.
 - Classification: Categorize text into different groups
 - Named entity recognition: Identify people, places, organizations in text

How to use them

- You give them a prompt (instructions or questions)
- They use everything they've learned to create new content

AWS Certified AI Practitioner (AIF-C01) Study Notes

- The better your prompt, the better the output
- You can guide them with examples, instructions, or conversation

Important characteristics

- Unpredictable: Same prompt might give different results each time
- Context window: Limited by how much text they can “remember” at once
- Knowledge cutoff: Only know information up to their training date
- Can hallucinate: Make up facts that sound believable but aren’t true
- No real understanding: They don’t truly understand meaning, just patterns

Popular LLM examples

- GPT-4 (OpenAI) - Powers ChatGPT
- Claude (Anthropic) - Known for being helpful and safe
- Gemini (Google) - Multi modal (can handle text, images, audio)
- Llama (Meta) - Open source, can run on your own hardware
- PaLM (Google) - Very large, used in various Google products

Generative Language Models

- These are models specifically designed to generate text that makes sense
- They predict what word comes next based on what came before (like autocomplete on steroids)

The way they work

- They break down text into smaller pieces called tokens
- Each token gets turned into numbers (embeddings) that the model can understand
- The model looks at all previous tokens and guesses the next one
- It builds responses word by word, considering the whole context

Example of next-word prediction

- Given: “After the storm passed, the village was”
- The model calculates probabilities for the next word:
 - “destroyed” (25% probability)
 - “flooded” (18% probability)
 - “quiet” (15% probability)
 - “empty” (12% probability)
 - “rebuilt” (8% probability)
 - ... and many other options
- The model randomly selects from these probabilities, so different runs might give different results
- Each choice affects the next word prediction, building a coherent story

Key characteristics

- Context-aware: They remember what you said earlier in the conversation
- Probabilistic: They don’t always pick the “best” word, they pick from likely options
- This is why you get different outputs for the same input

How they’re trained

AWS Certified AI Practitioner (AIF-C01) Study Notes

- Pre-training: Learn general language patterns from tons of text data
- Fine-tuning: Adjust the model for specific tasks (like following instructions)
- This two-step process makes them both knowledgeable and helpful

The power of generative models

- They can keep conversations going naturally
- They adapt their style based on your prompt
- They can be creative when asked (write stories, poems, code)

GenAI for Images

- Just like LLMs generate text, some models can generate images from text descriptions
- One popular type is called Diffusion Models (like Stable Diffusion from Stability AI)
- They basically learn what images look like and can create new ones based on your description

How diffusion models work

- Start with random noise (like static on a TV)
- Gradually remove noise step by step
- Guide the process with your text prompt
- End up with a clear, meaningful image

What they can do

- Text-to-image: Turn descriptions into images (“a red sunset over mountains”)
- Image-to-image: Modify existing images based on text (“make this photo look like a painting”)
- Inpainting: Fill in missing parts of images
- Style transfer: Apply artistic styles to photos

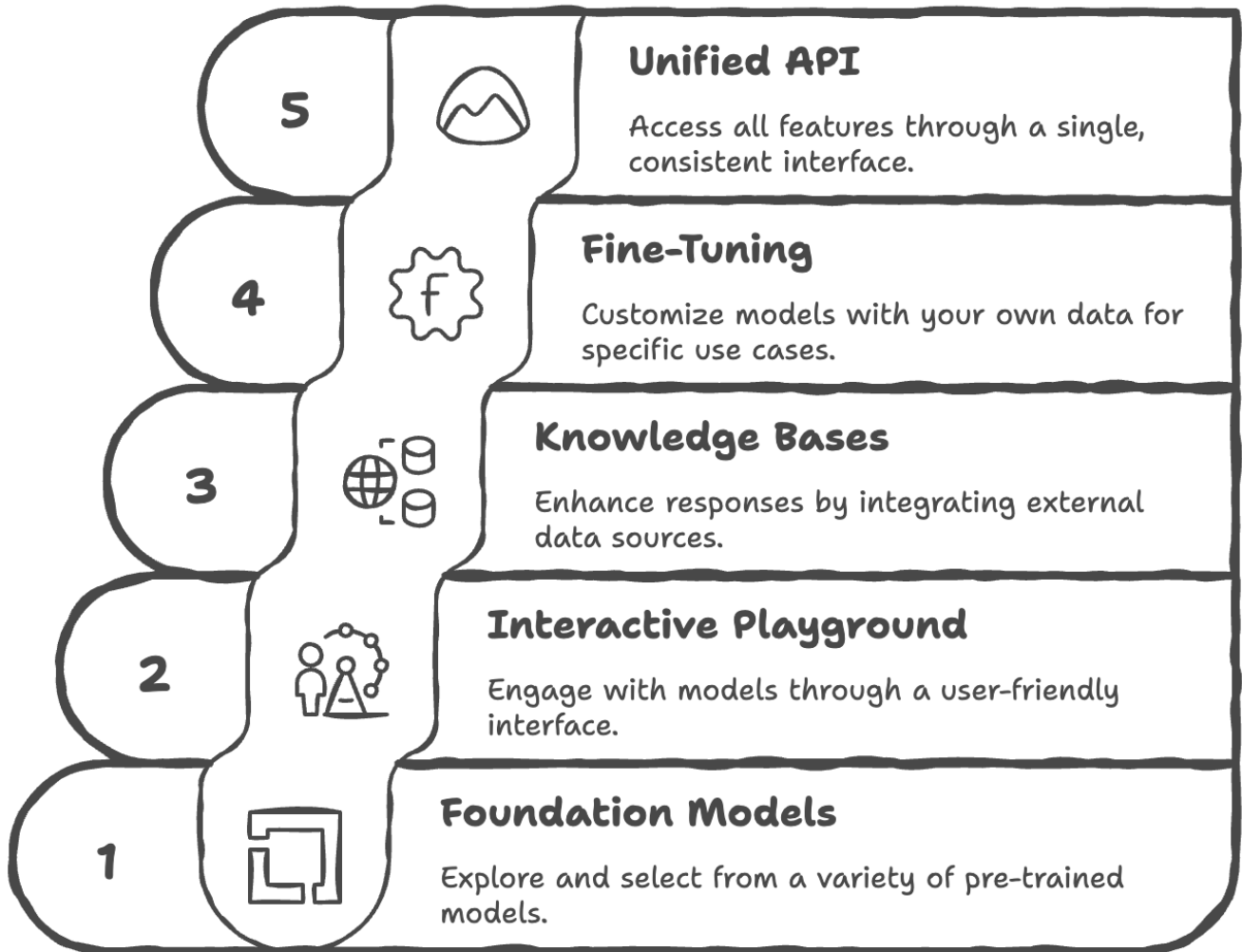
- Popular models and tools:
 - Stable Diffusion (open source, can run on your computer)
 - DALL-E (from OpenAI, used in ChatGPT)
 - Midjourney (very artistic, high-quality results)
- Key differences from LLMs:
 - Images are 2D grids of pixels (not sequential like text)
 - Need to learn patterns like colors, shapes, textures, lighting
 - Much larger files (images vs text)
 - More computational power needed
 - Usually slower to generate than text
- Common use cases:
 - Creating art and illustrations
 - Generating product mockups
 - Architectural visualization
 - Creating marketing visuals
 - Personalized avatars and portraits
 - Concept art for games and movies

Amazon Bedrock

What is Amazon Bedrock?

- **Primary service on AWS** for building generative AI applications
- A fully managed service with no infrastructure or service management required
- **Data Control & Privacy**: All data remains within your AWS account and never leaves—ensuring complete privacy and security
- Operates on a **pay-per-use pricing model** (no upfront commitment)
- Provides a **unified standardized API** for consistent access across all models
- Leverages a **wide array of foundation models** from multiple providers
- Offers advanced out-of-the-box features:
 - Retrieval-Augmented Generation (RAG)
 - LLM Agents
 - Knowledge Bases
- Built-in capabilities for **security, privacy, governance, and responsible AI**
- Users focus on applications while AWS manages the infrastructure

Amazon Bedrock



Foundation Models

- Amazon Bedrock hosts foundation models from multiple providers with AWS agreements:
 - AI21 Labs
 - Cohere
 - Stability.ai
 - Amazon
 - Anthropic
 - Meta
 - Mistral AI
 - Reference: <https://docs.aws.amazon.com/bedrock/latest/userguide/models-supported.html>
- Additional providers and models continue to be added over time
- **Model Isolation:** When you select a foundation model, Amazon Bedrock creates an exclusive copy accessible only within your account
- **Fine-tuning Support:** Some models can be fine-tuned with your own data for specific use cases
- **Data Privacy Guarantee:** None of your data is sent to model providers for training the original foundation models

AWS Certified AI Practitioner (AIF-C01) Study Notes

—all operations occur solely within your account

- The model copy can be customized to better suit specific needs without affecting the original foundation model

How to choose a foundation model?

Key Selection Factors

The choice of foundation model depends on multiple factors:

- **Model Type & Performance:** Different models excel at different tasks
- **Capabilities:** What the model can and cannot do
- **Constraints & Compliance:** Business and regulatory requirements
- **Customization Options:** Ability to fine-tune with your own data
- **Model Size & Inference Capabilities:** How outputs are generated (affects speed and resource usage)
- **Licensing Agreements:** Commercial use restrictions or requirements
- **Context Window:** Maximum amount of data you can send to the model (important for large documents/codebases)
- **Latency:** Response speed and performance
- **Multimodal Support:** Can the model handle multiple input types (text, audio, video, images) and produce multiple output types?

Important Considerations

- **Trade-offs:** Smaller models are more cost-effective but have less knowledge and capability
- **Testing is Essential:** There is no definitive answer; you must test models with your specific inputs and use cases
- **Language Support:** Different models support different languages
- **Cost Accumulation:** AI usage costs can accumulate rapidly, so pricing is a key consideration

Model-Specific Details

Amazon Titan

- High-performing foundation model directly from AWS (important for AWS certification)
- Supports multimodal inputs (text, images)
- Customizable through fine-tuning with your own data
- Cost-effective option
- Best for: Content generation, text classification, educational use cases

Llama-2 (Meta)

- Good for large-scale tasks and dialogue
- More expensive than Amazon Titan but cheaper than Claude
- 4,000 token limit (moderate context window)
- Best for: Technical content generation, customer service applications

Claude (Anthropic)

- Excellent for analysis, forecasting, and detailed document comparison
- **Largest context window (200,000 tokens)** - can process entire books, large codebases
- Most expensive option
- Best for: Complex analysis tasks, research, large document processing