How to Choose an Approach for Deploying Generative AI

Published 7 July 2023 - ID G00794559 - 14 min read

By Analyst(s): Arun Chandrasekaran, Leinar Ramos, Rajesh Kandaswamy

Initiatives: Digital Future; Artificial Intelligence; Drive Quantifiable Value With D&A Solutions for the Business

The popularity of ChatGPT has sparked interest in the adoption of generative AI; however, to make informed decisions and derive value, CTOs need to understand the various approaches. Here, we compare the deployment approaches and provide a decision framework for choosing one over the other.

Overview

Key Findings

- The pace at which generative AI providers are launching new capabilities is overwhelming IT leaders.
- The number of pretrained generative AI models and applications is growing exponentially; however, steering them to align with enterprise use cases and AI governance is challenging.
- Technology innovation leaders don't fully understand the variety of generative AI deployment approaches, and their pros and cons.
- Consuming models as embedded applications, embedding model APIs and steering them via prompt engineering are currently popular; extending them via a retrieval augmented generation architecture is an emerging approach.

Recommendations

Enterprise architecture and technology innovation leaders responsible for their enterprises' digital futures should:

- Understand and document the technical differences between each approach, so that they aren't locked in to the approaches prescribed by their vendors. They need to fully grasp the shared responsibility models with their vendors.

- Analyze the pros and cons of each deployment approach to bring clarity to their decision making. Align use cases with these approaches to ensure the right fit for each use case.
- Account for all critical decision factors and make objective decisions on a use-case-by-use-case basis. Approaches detailed here aren't mutually exclusive — most organizations may adopt a combination of these approaches.
- Monitor emerging trends to future proof their generative AI strategies — this is a rapidly evolving landscape, which calls for updating these strategies every few months.

Strategic Planning Assumptions

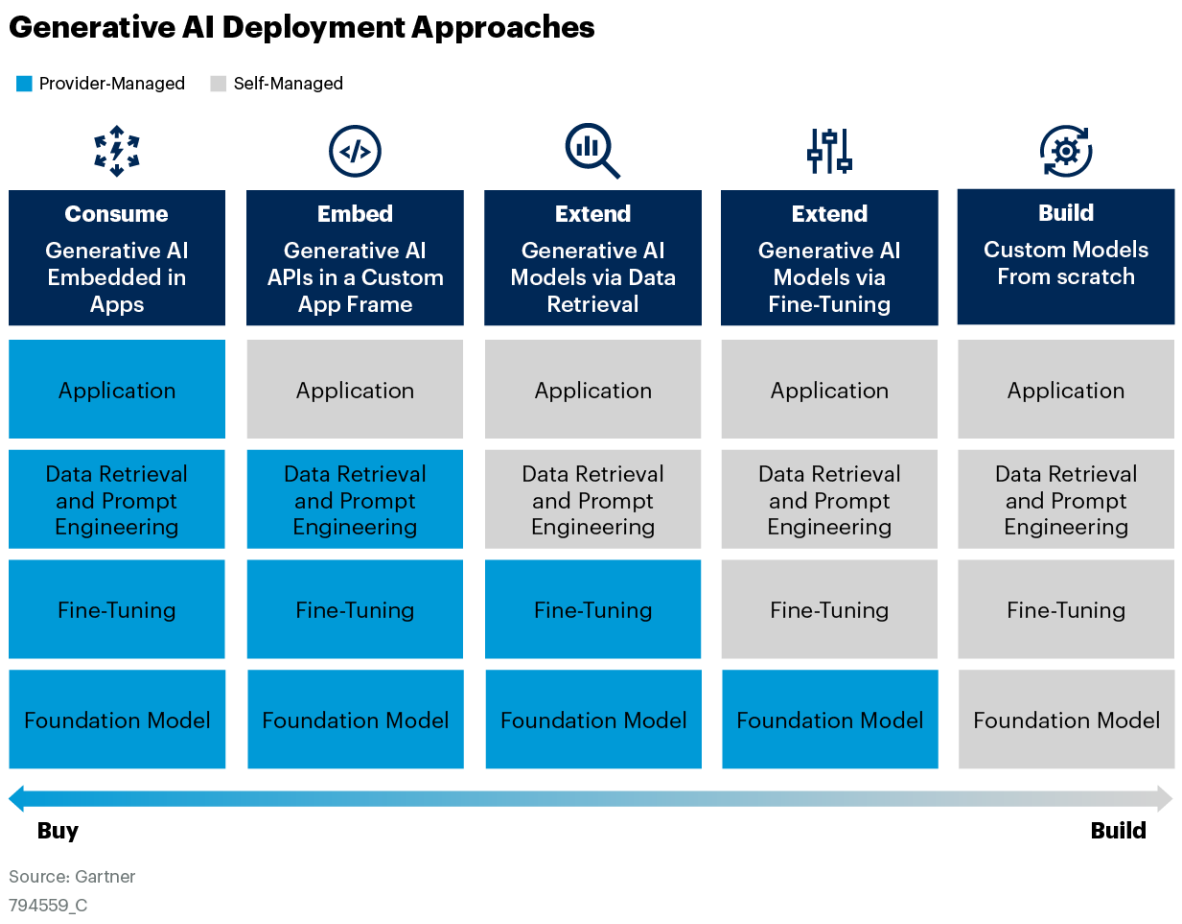
- By 2026, more than 80% of enterprises will have used generative, artificial intelligence (AI) APIs, models and/or deployed generative, AI-enabled applications in production environments, which is a significant increase from fewer than 5% today.
- By 2026, more than 70% of independent software vendors (ISVs) will have embedded, generative AI capabilities in their enterprise applications, which is a major increase from fewer than 1% today.
- By 2026, nearly 80% of prompting will be semiautomated — through automated prompting tools or via autonomous agents, which require limited prompting.
- By 2028, more than 50% of enterprises that have built their own models from scratch will abandon their efforts due to costs, complexity and technical debt in their deployments.

Introduction

The popularity of ChatGPT has opened the floodgates of innovation in the generative AI space. The past six months has seen a flurry of AI foundation models, provider fine-tuned models, generative AI applications and machine learning (MLOps) tools released in the market. In addition, many large incumbent ISVs are embedding generative AI into their existing applications to bring the power of generative AI to business users. Although these fast-paced developments signify the competitive jostling that is characteristic of most high-stakes, early-stage markets, this also presents a confusing array of choices that enterprise IT leaders need to navigate. Here, we aim to demystify the differences between various generative AI deployments approaches (see Figure 1). and outline a decision framework for choosing one over the other.

Analysis

Figure 1: Generative AI Deployment Approaches



The simplest way to deploy generative AI is by consuming applications such as ChatGPT through their web interfaces or mobile apps. However, those are more-consumer-oriented services, although OpenAI and others plan to introduce enterprise grade services (e.g., ChatGPT for Business) in the future, with better data privacy terms. Beyond this, Gartner sees five key approaches to consuming generative AI capabilities emerging:

1. **Consume generative AI embedded in applications:** Organizations can directly use commercial applications that have generative AI capabilities embedded in them. An example of this would be using an established design software application, which now includes image generation capabilities (e.g., Adobe Firefly).

2. **Embed generative AI APIs in a custom application frame:** Enterprises can build their own applications, integrating generative AI via foundation models APIs. Most closed-source generative AI models (GPT-3, GPT-4, PaLM 2, etc.) are available for deployment via cloud APIs. This approach can be further refined by prompt engineering — this could include templates, examples — to better inform the foundation model output. An example of this is sentiment analysis, where you can provide detailed instructions in the prompt on how you want the sentiment to be classified (e.g., positive, negative or neutral) and illustrate it with examples to steer the models in providing the optimal response.

3. **Extend Generative AI models via data retrieval:** Retrieval augmented generation (RAG) enables enterprises to retrieve data from outside a foundation model (often your internal data) and augment the prompts by adding the relevant retrieved data. This will improve the accuracy and quality of model response for domain-specific tasks. RAG doesn't require the creation of custom models. An example is searching a private document database to find relevant data to add into the foundation model's prompt, augmenting its response with relevant, similar information.

4. **Extend generative AI models via fine-tuning:** Fine-tuning takes a large, pretrained foundation model as a starting point and further trains it on a new dataset to incorporate additional domain knowledge or improve performance on specific tasks. This often results in custom models that are dedicated to the organization. For example, an insurance company could fine-tune a foundation model with its own policy documents to incorporate this knowledge into the model and improve its performance on specific use cases.

5. **Build custom foundation models from scratch:** Organizations could ultimately build their own foundation models from scratch, fully customizing them to their own data and business domains. For example, a financial institution might create a foundation model trained with financial data, which could then be used for many financial services use cases (BloombergGPT is a prominent example).

There is no single best-approach for all use cases, but real trade-offs between these approaches need to be considered when making informed decisions.

Analyze the Pros/Cons of Each Deployment Approach to Bring Clarity to Your Decision-Making

Consume Generative AI Embedded in Applications

Pros

- Easier to deploy, with low or no fixed costs required to start experimenting with generative AI capabilities.
- Improvements to the underlying generative AI model powering the application could be directly translated to increased utility for users, without an additional investment.
- Easy integration with existing workflow — making this the least-disruptive approach.

Cons

- Lack of flexibility to build more-complex workflows that are not part of the standard application features and the inability to extend the model capabilities beyond that application.
- Applications with embedded AI may not be able to deeply understand the context of a conversation or task, leading to less-accurate or less-relevant responses.
- Organizations have less control over security and data privacy risk, with a strong dependence on the application provider's security and data protection controls, which could widely vary, depending on the type of software provider.

Embed Generative AI APIs in a Custom Application Frame

Pros

- Easier to implement and with lower fixed costs, because you pay for the use of the model (inference) only, not for its training. This approach often provides time-to-market advantages — you can get your use case to production faster, with acceptable degrees of customization.
- Foundation models have proved to be effective few-shot learners, which means, with a limited number of high-quality samples, they can complete new tasks with adequate accuracy.
- In prompt engineering, the underlying foundation model is frozen — this provides the ability to use the same model across a variety of use cases — which simplifies model management and governance.

Cons

- There is a limit to how much data can be transmitted via prompts, which can limit use cases for this approach.

- Prompt engineering is a nascent field, where best practices are only emerging, and for which new skills are required.
- Potential backward-compatibility issues, with underlying API model changes affecting the custom workflows and applications built on top of them. Vendors must continue to support previous versions of the models.

Extend Generative AI Models via Data Retrieval

Pros

- Data retrieval via RAG enables organizations to incorporate additional information beyond what was in the foundation model's training data. This could be more up-to-date data, as well as domain-specific or private data.
- Extending the models via a RAG approach can provide an appropriate balance between bringing organizational context into foundation models without the complexity and cost of modifying the underlying models (fine-tuning or building models from scratch).
- Improved accuracy on domain-specific tasks and reduced hallucinations in model output due to prompt enhancement and grounding on the organization's data.

Cons

- A RAG approach is limited by the context window of the generative model, constraining the amount of retrieved information that can be sent to the model. Similarly, the additional retrieval step to augment the prompt may increase latency, impacting its viability for real-time use cases.
- Implementing a RAG approach involves redesigning the technical architecture and workflow to include new technology components such as vector databases and embedding models — the know-how about these technology components and the overall architecture is pretty rudimentary in most enterprises. These additional components carry additional costs.
- Organizations need to ensure that no sensitive or privileged information is leaked via the retrieval process, which entails adequate guardrails around access control, information retrieval and monitoring the retrieval output.

Extend Generative AI Models via Fine-Tuning

Pros

- Fine-tuning enables organizations to quickly improve performance for specific use cases, without having to train a full model from scratch. This can result in improved performance and the ability to reduce hallucinations by fine-tuning it with organizational data and/or domain-specific data for particular tasks.
- Fine-tuning typically doesn't require large amounts of data (although the data needs to be high-quality), particularly when contrasted with the data required to train the underlying foundation models, which would require orders of magnitude more data.
- There is an ongoing trend toward smaller, but high-performing, open-source foundation models. This could make the creation of fine-tuned models more feasible and cost-effective.

Cons

- The cost of using a fine-tuned model (inference cost) can be significant, even if the cost of fine-tuning training is not high. The fine-tuned models are still large models, with billions of parameters, and would need to be optimized to realistically be used at scale.
- Fine-tuning on top of a given foundation model might restrict future flexibility in the use of better foundation models that may emerge.
- Foundation models fine-tuned for specific use cases might lose their ability to be extended to broader use cases. Fine-tuning might overspecialize models for specific use cases, losing their more-general abilities.

Build Custom Foundation Models From Scratch

Pros

- The models are fully customized to your use case or domain — theoretically, this approach has the potential to deliver the highest accuracy.
- If adequate data governance is in place, then the organization will have complete control over the training datasets and model parameters. This can significantly increase use-case alignment, and reduce bias and other unintended or harmful consequences.
- Complete control over your model can yield competitive differentiation and a strong product offering, relative to your peers. The foundation model could be potentially commercialized if it is sufficiently high-performing and domain-specific.

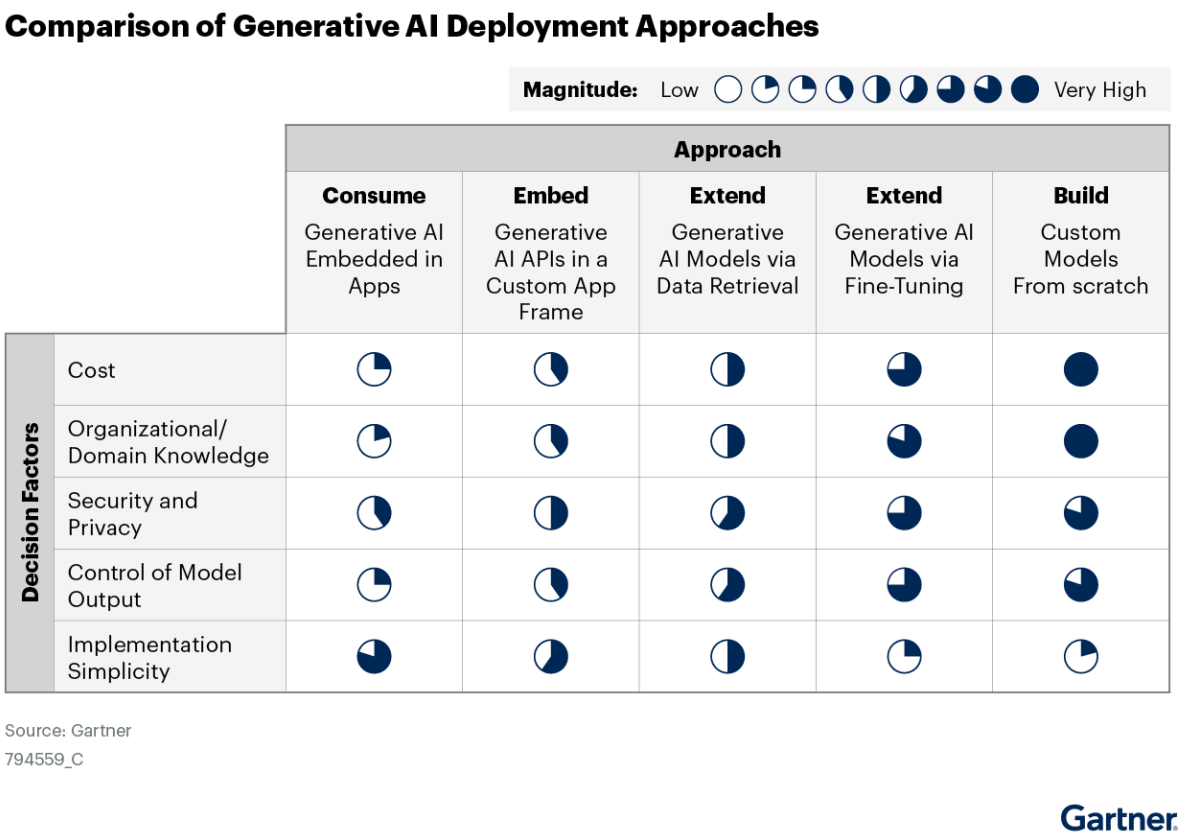
Cons

- The cost of training and maintaining a large, generative AI model can be high. It can include training infrastructure costs, data acquisition, infrastructure and labeling costs, human audit of the model quality, and the inferencing costs.
- Continued access to cutting-edge AI researchers is important for building a high-quality model and regularly maintaining and updating it.
- The pace of innovation in the generative AI ecosystem is fast and, for most clients, the pace of external innovation from technology vendors will be greater than their own internal innovation. In the future the build decision may induce regret.

Use This Decision Framework to Compare Approaches on a Use-Case Basis

The approaches described above aren't mutually exclusive. Most organizations will deploy a few, if not all, of them, depending on the use case, technical know-how, maturity of the organization and time-to-market needs. Figure 2 provides a decision framework on some of the key decision factors to consider while making decisions on which deployment approach should be the preferred approach for a specific use case.

Figure 2: Comparison of Generative AI Deployment Approaches



The key factors to consider are:

- **Costs** — Consuming embedded applications and embedding model APIs incurs no training costs, because you are consuming pretrained models "as is," but still incur usage costs (either on a subscription or pay-per-use basis). When you do prompt engineering, each time you modify your human-created prompt, you only incur inference costs without updating the model parameters, so the cost is low. Data retrieval increases inference costs by adding more data to the model prompts. Fine-tuning costs could vary widely based on the size of the model — costs would be high when updating models with billions of parameters. Building a model from scratch would be the most expensive of these approaches.

- **Organizational and Domain Knowledge** — Most AI foundation models are versatile, but general-knowledge models. Hence, bringing use-case-specificity or domain-specificity is important in improving the accuracy and reducing the unintended consequences of using these models. Embedding model APIs via data retrieval, fine-tuning and building your own models offer the best possibilities to inject the organizational or domain knowledge to these models. More providers (e.g., Charli AI in financial services, Huma AI in life sciences and BloombergGPT) offer domain-trained models and SaaS applications, which can also bring the general-purpose models closer to organizational needs.

- **Ability to Control Security and Privacy** — Security and privacy considerations are quite broad in the generative AI market. They include protection of a company's intellectual property (IP), control over its data, protection against legal liability, need for ownership of models, applications and model output, adequate access control, and data sovereignty. Building your own models or creating custom models via fine-tuning provides stronger ownership of key assets and more flexibility in terms of the controls you can implement. The providers' privacy and security controls are rapidly evolving, but it is incumbent on the user organization to understand the user versus provider shared responsibilities. Ensure they have an audit process to weed out poorly secured and architected generative AI products.

- **Control of Model Output** — Model quality is an important factor. An AI foundation model, while a significant technology advancement, is prone to hallucination risks, as well as propagating biased or harmful behavior. Consuming the models "as is" via an ISV may not be an effective approach for a lot of client-facing use cases, unless there is scope for prompting. Hence, data retrieval, model fine-tuning and building your own models might be preferred in high-control environments. Generative AI is stochastic and can never be relied on for 100% accuracy. Business-critical applications will require a human in the loop.

- **Implementation Simplicity** — Consuming embedded applications and embedding model APIs have advantages, due to their inherent simplicity and time to market. They don't have a significant negative impact in terms of current workflows.

Monitor the Emerging Trends to Future Proof Your Strategy

Although the approaches described above capture the zeitgeist of approaches, it isn't future-proof, due to the rapid evolution of generative AI. Other emerging approaches include the following.

Open-Source

Although closed-sourced generative AI models dominate the landscape today, open-source models are rapidly evolving. The rise of open-source models expands the choice of foundation and fine-tuned models available to enterprises and through model hubs, such as Hugging Face, they make it easier for developers to experiment with and iterate models. Technology companies such as Meta, Databricks, Hugging Face, Stability AI and EleutherAI, as well as academic institutions, are all significant contributors to open, generative AI models.

Prompt Tuning

Prompt tuning as an approach lies between prompt engineering and model fine-tuning. In prompt tuning, prompts are fed into the model to give it task-specific context. This input provided to the model (in the form of prompts) will enable it to perform well across specific tasks, without requiring significant training data. Prompt tuning may potentially offer good task performance, while keeping the pretrained model frozen, enabling efficient multitask serving. Prompt tuning requires more data and generally has higher costs than prompt engineering. Prompt tuning is a nascent approach. This blog from Google research provides further details on this approach — Guiding Frozen Language Models With Learned Soft Prompts.

Agents

The foundation models that power generative AI are limited, and they require tools that can harness their potential effectively. More complex agents can be built to provide a variety of capabilities including:

- Working toward a goal through tasks and subtasks
- Enabling business workflows for complex work
- Retrieving and using information from other sources, such as databases, filesystems and the web
- Effectively using prompts and the outputs to achieve the objectives
- Automation of all of the above, including error handling and fault tolerance

Agents that address some combination of the above are emerging, both commercially and in open source. Examples include LangChain, LlamaIndex, AutoGPT and BabyAGI. We expect more to emerge, because advances in this field will help maximize the potential of generative AI. However, caution is warranted, because these tools are nascent and most are not robust enough to control the underlying issues of foundation (e.g., hallucinations and inaccurate input). They can amplify such issues and make them worse. For example, an automatic agent that uses foundation models to achieve goals recursively by breaking them into tasks can lead to completely wrong outcomes and irrevocable and harmful business decisions.

Evidence

Detailed interviews were conducted with several generative AI vendors and research labs.

Recommended by the Authors

Some documents may not be available as part of your current Gartner subscription.

Innovation Insight for Artificial Intelligence Foundation Models

Innovation Insight for Generative AI

AI Design Patterns for Large Language Models

How to Pilot Generative AI

Gartner Addresses Frequently Asked Questions on ChatGPT

An Early Look at Corporate Guidance on ChatGPT
