Building AI-Driven Spring Applications

With Spring AI

Sandra Ahlgrimm
Senior Cloud Advocate, Microsoft

Timo Salm
Senior Lead Tanzu DevX Solution Engineer, Broadcom

July 2024

Microsoft & vmware by Broadcom

Who We Are

Sandra Ahlgrimm

Senior Cloud Advocate
Microsoft

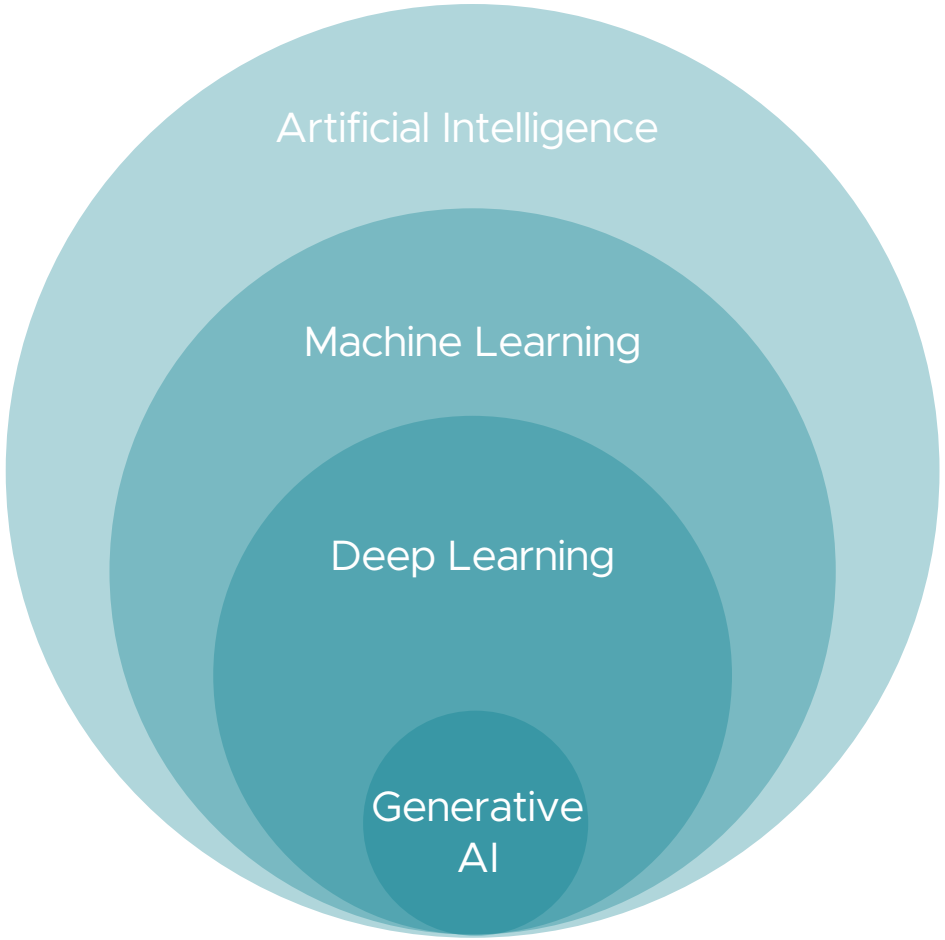
X (Twitter): @sKriemhild
LinkedIn: https://linkedin.com/in/sandraahlgrimm

Timo Salm

Senior Lead Tanzu DevX Solution Engineer – EMEA
Broadcom

X (Twitter): @salmto
LinkedIn: https://linkedin.com/in/timosalm

(Generative) AI Fundamentals



Artificial Intelligence: Machines are capable of performing cognitive functions typically associated with human minds

Machine learning: Algorithms that learn from data to make predictions or decisions without being explicitly programmed.
Types of ML: Supervised, Unsupervised, Reinforcement Learning, …

Deep Learning: Algorithms that simulate how the human brain's neurons work (Neural Networks)

Generative AI is capable of generating text, images, or other data by utilizing models that learn patterns and structure of their training data

Generative AI Fundamentals

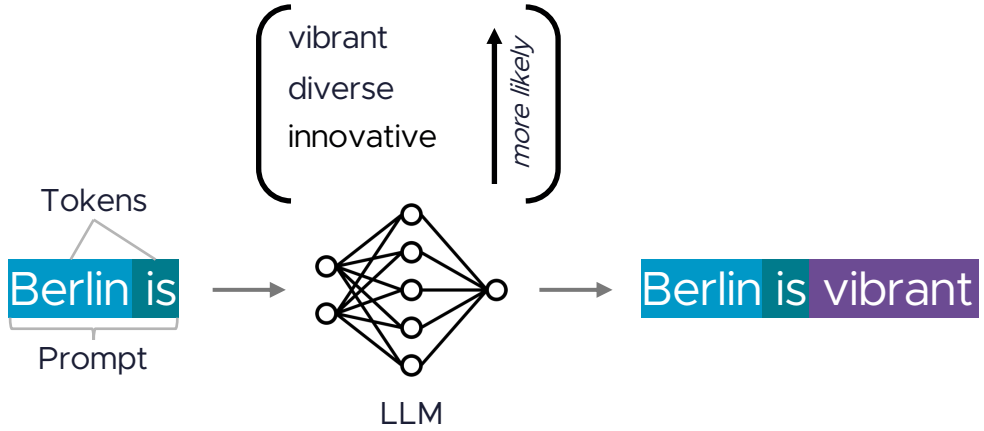
Machine Learning Model: A mathematical model trained on a specific dataset to make predictions or classifications on new data

Foundation Model: An ML model trained on a huge amount of generic data that serves as the base for various generative tasks

Large Language Models (LLMs): AI models specifically designed to understand and generate human language

- Prompts: Input instructions or data given to the model to guide content generation
- Tokens: Basic units of data processed by models, such as words or parts of words in text generation

How LLMs work

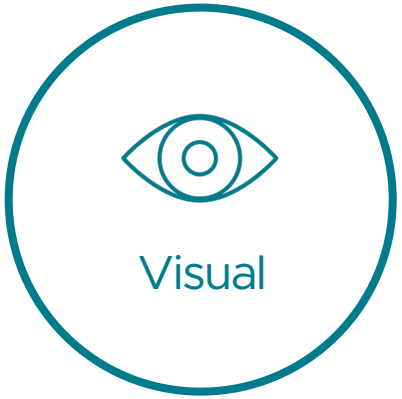


LLMs process a specific number of tokens at a time using complex mathematical calculations to predict the most likely next token in a sequence.

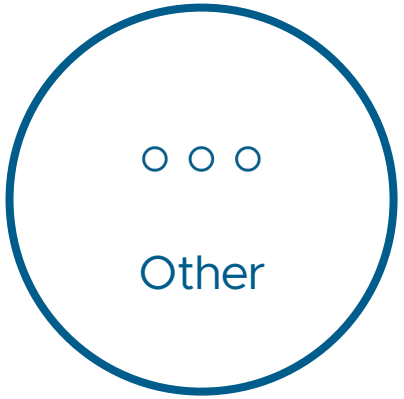
Generative AI Fundamentals

Use Case Examples

- General writing
- Translation
- Code generation
- Marketing content
- Customer support



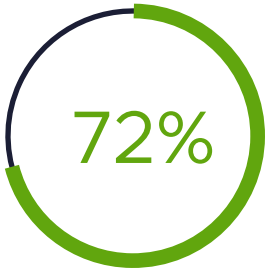
- Image/Video generation and editing
- Design
- 3D Models



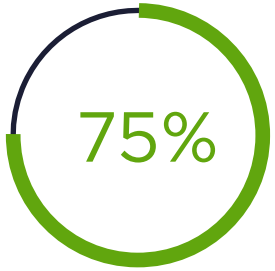
- Audio/Music/ Voice generation
- Biology and Chemistry
- Gaming

spring® by VMware Tanzu

The most popular application development framework on the JVM



of Java developers utilize Spring Boot



like Spring because it's very stable, scalable, and secure

History of Spring

Created in 2003 as a lightweight alternative to address the complexity of the early J2EE specifications. Spring and Java/Jakarta EE are not in competition, they are complementary.

Why Spring?

- Focused on speed, simplicity, and productivity
- Enterprise and production ready
- Extensive ecosystem
- Large, active developer community

Microsoft & vmware by Broadcom

Sources: https://www.jetbrains.com/lp/devecosystem-2023/java/
https://tanzu.vmware.com/content/ebooks/the-state-of-spring-2022

Official Spring Projects

spring.io/projects

Spring Framework

Provides core support for dependency injection, transaction management, web apps, data access, messaging, and more.

Spring Boot

Takes an opinionated view of building Spring applications and gets you up and running as quickly as possible.

Spring Session

Provides an API and implementations for managing a user's session information.

Spring Integration

Supports the well-known Enterprise Integration Patterns through lightweight messaging and declarative adapters.

Spring Data

Provides a consistent approach to data access – relational, non-relational, map-reduce, and beyond.

Spring Cloud

Provides a set of tools for common patterns in distributed systems. Useful for building and deploying microservices.

Spring HATEOAS

Simplifies creating REST representations that follow the HATEOAS principle.

Spring Modulith

Spring Modulith allows developers to build well-structured Spring Boot applications and guides developers in finding and working with application modules driven by the domain.

Spring Cloud Data Flow

Provides an orchestration service for composable data microservice applications on modern runtimes.

Spring Security

Protects your application with comprehensive and extensible authentication and authorization support.

Spring REST Docs

Lets you document RESTful services by combining hand-written documentation with auto-generated snippets produced with Spring MVC Test or REST Assured.

Spring AI

Spring AI is an application framework for AI engineering.

Spring Authorization Server

Provides a secure, light-weight, and customizable foundation for building OpenID Connect 1.0 Identity Providers and OAuth2 Authorization Server products.

Spring for GraphQL

Spring for GraphQL provides support for Spring applications built on GraphQL Java.

Spring Batch

Simplifies and optimizes the work of processing high-volume batch operations.

Spring CLI

A CLI focused on developer productivity

AND Spring AMQP, Spring Flo, Spring for Apache Kafka, Spring for Apache Pulsar, Spring LDAP, Spring Shell, Spring Statemachine, Spring Web Flow, Spring Web Services

Spring AI

An application framework for AI engineering

Provides the key ingredients for creating (Generative) AI applications!

Aligns with Spring ecosystem design principles:

- Component abstractions and default implementations
- Portability across AI Provider APIs, Vector databases and more
- Auto Configuration and Starters with Spring Boot

Additional features

- Support for several techniques to adapt models to your needs like “Function Calling” or “Retrieval Augmented Generation”
- Multimodality
- Easy conversions of model output into a structured format
- Evaluation Testing support
- Builds upon other projects in the Spring ecosystem

Demo

Getting Started with Spring AI

Microsoft & vmware by Broadcom

Demo

Switch to Model From Different Vendor

Microsoft & vmware by Broadcom

Adapting Foundation Models

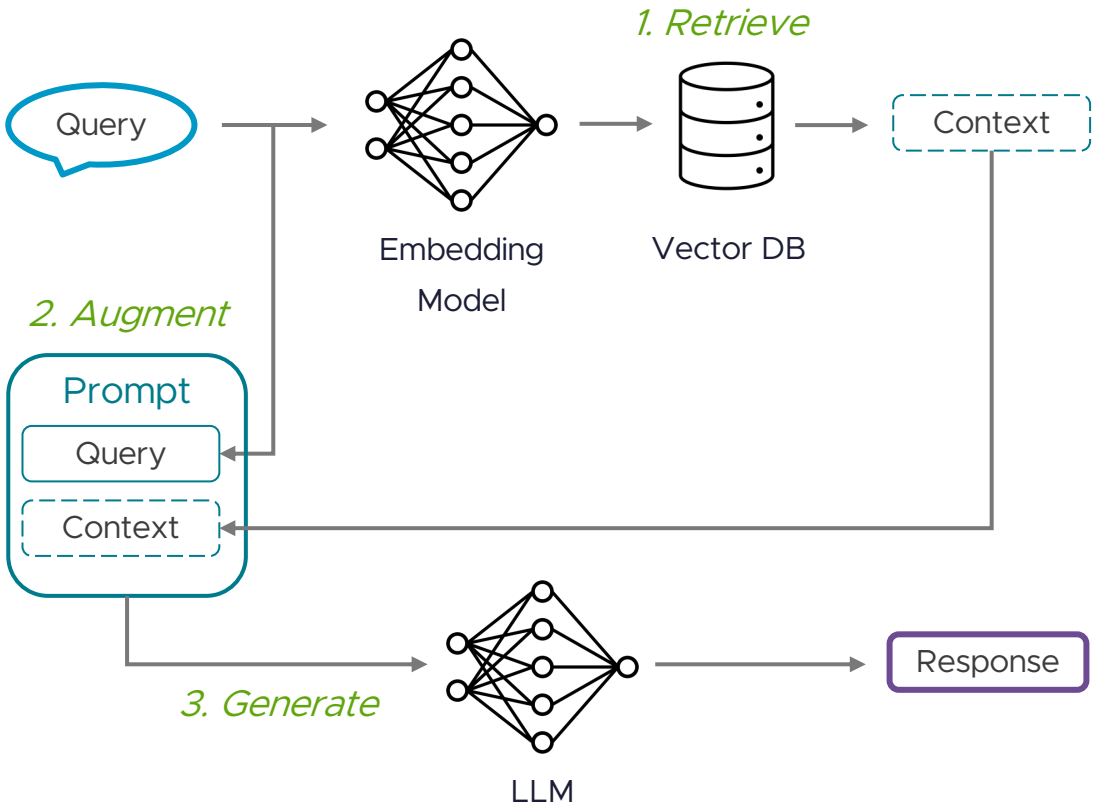
Popular techniques

Fine-Tuning: Further refining a model with specific data to improve its performance on a particular type of task ← *requires significant computational resources*

Prompt Engineering: Designing effective input prompts to guide a generative model's outputs (e.g. with Few-Shot Prompting, Chain-of-Thought Prompting, or In-Context Learning)

Function Calling: Allows you to register your own functions to connect the model to the APIs of external systems

Retrieval-Augmented Generation (RAG) enhances the output of models by incorporating relevant external information from external data sources



Anatomy of a RAG System

Adapting Foundation Models

With Spring AI

Function Calling

Spring AI handles the function invocation conversation for you:

- Provide your function as a `@Bean` that returns a `java.util.Function`
- Activate the function in your prompt options
- Multiple functions can be defined and referenced in a single prompt

Demo

Function Calling with Spring AI

Microsoft & vmware by Broadcom

Adapting Foundation Models

With Spring AI

Function Calling

Spring AI handles the function invocation conversation for you:

- Provide your function as a `@Bean` that returns a `java.util.Function`
- Activate the function in your prompt options
- Multiple functions can be defined and referenced in a single prompt

Retrieval-Augmented Generation

Spring AI provides:

- The Vector Store API provides portability across different vector database providers
- ETL(Extract Transform and Load) framework to populate the vector database with embeddings
- An Embeddings Model API with support for several models
- RAG will be performed by providing an instance of `QuestionAnswerAdvisor` to the `ChatClient` that queries the vector database for documents related to the user question

Demo

RAG with Spring AI

Microsoft & vmware by Broadcom

Follow us
@sKriemhild
@salmto

Thank You

https://github.com/timosalm/spring-ai

Microsoft & vmware by Broadcom