RAG Fundamentals

Introduction to Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation, commonly referred to as RAG, is an AI architecture that combines information retrieval with natural language generation.

Instead of relying solely on a model’s internal parameters, RAG systems retrieve relevant documents from an external knowledge base.

This approach improves factual accuracy, reduces hallucinations, and allows models to work with up-to-date or domain-specific data.

Core Components of a RAG System

A typical RAG system consists of a document store, an embedding model, a retriever, and a generator.

Documents are chunked and transformed into vector embeddings which are stored in a vector database.

At query time, relevant chunks are retrieved and passed to a language model for answer generation.

Document Chunking and Embeddings

Chunking is the process of splitting documents into smaller, semantically meaningful units.

Good chunk sizes balance context richness with retrieval precision.

Embedding models convert text into numerical vectors that capture semantic meaning.

Vector Databases and Similarity Search

Vector databases such as FAISS, Pinecone, and Weaviate are commonly used in RAG systems.

They enable efficient similarity search over large collections of embeddings.

Cosine similarity and dot product are popular distance metrics for retrieval.

Evaluation and Use Cases

RAG systems are evaluated using metrics such as retrieval accuracy, answer relevance, and faithfulness.

Common use cases include chatbots, internal knowledge assistants, and question answering systems.

RAG is especially useful in enterprise and regulated environments.