

April 27, 2025

InvestSage: Intelligent Investment Advice



Agenda

- InvestSage: Intelligent Investment Advice with RAG & Akka
- The Challenge in Investment Advice
- How InvestSage Understands You Better: RAG
- Finding the Right Context Fast
- InvestSage RAG Flow
- InvestSage in Action
- Handling Investment Data Efficiently
- The Akka Embedding Pipeline
- Configurable & Managed Processing
- Why InvestSage Stands Out
- Thank You

InvestSage: Intelligent Investment Advice with RAG & Akka

- InvestSage combines Retrieval-Augmented Generation (RAG) with Akka for real-time, intelligent investment advice.
- RAG enriches AI responses with relevant, up-to-date financial data, reducing errors and hallucinations.
- Akka provides a resilient, high-concurrency backend for efficient data processing and embedding.
- The solution integrates cutting-edge AI models with a robust architecture for scalability and performance.
- Designed for financial professionals and investors seeking reliable, data-grounded insights.

The Challenge in Investment Advice

Limitations of Generic AI

Standard large language models lack access to the latest financial data and domain-specific knowledge, leading to generic or outdated investment advice.

Risks of Inaccurate Suggestions

Without real-time context, AI can 'hallucinate' facts or provide misleading guidance, which is critical to avoid in financial decision-making.

InvestSage's Solution Approach

Combines Retrieval-Augmented Generation (RAG) with a scalable Akka-based data processing backend to deliver precise, context-aware investment insights.

How InvestSage Understands You Better: RAG

Retrieve Relevant Information

InvestSage first searches its curated investment knowledge base to find pertinent information related to the user's query, ensuring that responses are grounded in accurate and up-to-date data.

- Relevant investment data chunks
- Indexed knowledge base
- Efficient vector search results



Augment AI Prompt

The retrieved information is combined with the user's original question to create an augmented prompt that provides context, improving the AI's understanding and response quality.

- Augmented prompt
- Contextual data integration
- Enhanced query input

Generate Informed Answer

The augmented prompt is fed into a large language model (LLM) which generates a detailed, accurate, and context-aware answer tailored to the user's specific investment inquiry.

- Contextual AI response
- Detailed investment advice
- Reduced misinformation

Deliver Reliable Guidance

InvestSage provides the user with a grounded and trustworthy response, minimizing guesswork and outdated or hallucinated information through the RAG process.

- Accurate investment recommendations
- Trustworthy AI interactions
- Improved user confidence

Finding the Right Context Fast



Embeddings Explained

Investment texts are transformed into numerical vectors called embeddings using AI models like OpenAiEmbeddingService, capturing semantic meaning for accurate comparisons.



Similarity Search Process

User queries are embedded similarly, then a fast cosine similarity search identifies the closest matching data chunks from the investment embeddings table.



Vector Database Storage

These embeddings are stored in PostgreSQL enhanced with the pgvector extension, enabling efficient and scalable handling of high-dimensional vector data.



RagDataProcessor Role

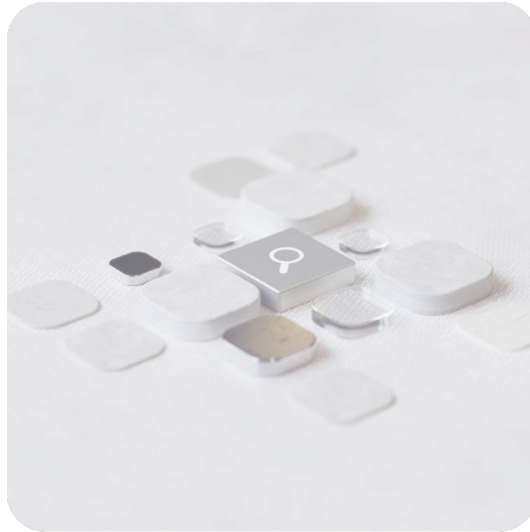
This component orchestrates embedding queries and similarity searches, ensuring rapid retrieval of relevant investment information for prompt AI augmentation.

InvestSage RAG Flow



User Query Processing

User inputs are embedded using OpenAiEmbeddingService to create a vector representation for similarity search.



Vector Search & Context Retrieval

RagDataProcessor performs a fast similarity search in PostgreSQL pgvector database to find relevant investment data chunks.



Augmented Prompt & LLM Generation

Relevant context is combined with the user query to build an augmented prompt, then GPT-4.1-nano generates the final response.

InvestSage in Action

Live Demo Interaction

- User inputs: 'What are the main risks of investing in emerging markets?'
- InvestSage first retrieves relevant investment data chunks to enhance context.
- The AI generates a detailed answer grounded in current financial information.
- Response highlights risks like political instability, currency fluctuations, and market volatility.

Behind the Scenes: Code Highlights

- `findSimilarChunks` retrieves relevant data from embeddings for context.
- `buildAugmentedPrompt` combines retrieved data with user query for input to LLM.
- `getChatResponse` calls the language model to generate the final, informed reply.
- This pipeline ensures accuracy and contextual relevance in responses.

Handling Investment Data Efficiently



The Challenge

Processing large volumes of complex financial text data demands a robust and scalable system to ensure timely and accurate embeddings for RAG.



Our Approach: Akka Actor System

Utilizing Akka's distributed actor model to enable parallel processing, high concurrency, and resilience in handling data chunking and embedding tasks.



Benefits of Akka Pipeline

Accelerates data ingestion and embedding, scales horizontally by adding resources, and provides fault tolerance to maintain system reliability under load.

The Akka Embedding Pipeline



Data Reading

Multiple `DataReaderActors` read raw investment data in parallel, batching it for efficient processing.



Chunking & Embedding



`ChunkerActor` splits raw data into smaller chunks; `EmbeddingActors` convert chunks into vector embeddings by calling OpenAI API.



Database Writing & Coordination

`DbWriterActors` write embeddings to PostgreSQL; `JobCoordinatorActor` manages workflow, error handling, and lifecycle.

Configurable & Managed Processing

- **Fine-tuning Concurrency for Performance**
 - Configure number of DataReaderActors (N) to control input throughput.
 - Adjust EmbeddingActors (P) to match API call capacity and speed.
 - Set DbWriterActors (M) to optimize database write operations.
 - Scalable settings allow resource-based tuning for different deployment environments.
- **Managed Lifecycle and Reliability**
 - JobCoordinatorActor oversees the entire pipeline lifecycle.
 - Handles startup sequences to ensure proper initialization.
 - Manages orderly shutdown to maintain data consistency.
 - Implements error handling and recovery strategies.
 - Ensures fault tolerance and robust processing.

Why InvestSage Stands Out

Accurate & Relevant

RAG technology ensures answers are grounded in up-to-date, curated investment data, eliminating hallucinations and guesswork.

Scalable Architecture

Akka's actor system enables high concurrency and fault tolerance, allowing InvestSage to efficiently process large volumes of financial data.

Fast Similarity Search

pgvector extension in PostgreSQL enables lightning-fast vector similarity searches for retrieving relevant investment context quickly.

Modern Technology Stack

Built with Java 17, Spring Boot 3+, Spring AI, Akka, and PostgreSQL, ensuring robust, maintainable, and future-proof development.

Maintainable & Configurable

Clean architecture and configurable pipeline parameters allow easy tuning and long-term maintainability of the system.

Thank You

InvestSage offers a scalable and intelligent solution for reliable investment advice.

Powered by Retrieval-Augmented Generation and a robust Akka backend, it combines accuracy with performance.

Thank you for your attention and interest in advancing investment guidance technology.

Thank you.



