REVISTA DIGITAL ACADÊMICO MUNDO

MELHORES ARQUITETURAS DE SOFTWARE PARA SISTEMAS COM INTELIGÊNCIA ARTIFICIAL

BEST SOFTWARE ARCHITECTURES FOR SYSTEMS WITH ARTIFICIAL INTELLIGENCE

LAS MEJORES ARQUITECTURAS DE SOFTWARE PARA SISTEMAS CON INTELIGENCIA ARTIFICIAL

10.56238/ramv20n16-018

Carlos Claudio Pereira da Silva
Graduando em Engenharia de Software
Instituição: Fundação Centro de Análise, Pesquisa e Inovação Tecnológica (FUCAPI)
Endereço: Amazonas, Brasil
E-mail: claudiopereira6546365463@gmail.com

Alexandre Castro
Orientador
Instituição: Fundação Centro de Análise, Pesquisa e Inovação Tecnológica (FUCAPI)
Endereço: Amazonas, Brasil

RESUMO
O avanço acelerado da Inteligência Artificial (IA) tem exigido das equipes de engenharia de software a adoção de arquiteturas robustas, escaláveis e observáveis para sustentar modelos de aprendizado de máquina em produção. Este trabalho apresenta uma análise comparativa das principais arquiteturas de software utilizadas atualmente no desenvolvimento e implantação de sistemas baseados em IA, abordando padrões como microsserviços, serverless, pipelines de MLOps, arquitetura orientada a eventos e sistemas RAG (Retrieval-Augmented Generation). São discutidos os trade-offs de cada abordagem, critérios de escolha e tendências emergentes para 2025 e além.

Palavras-chave: Inteligência Artificial. Arquitetura de Software. Microsserviços. MLOps. RAG.

ABSTRACT
The rapid advancement of Artificial Intelligence (AI) has demanded that software engineering teams adopt robust, scalable, and observable architectures to support machine learning models in production. This work presents a comparative analysis of the main software architectures currently used in the development and deployment of AI-based systems, addressing patterns such as microservices, serverless, MLOps pipelines, event-driven architecture, and RAG (Retrieval-Augmented Generation) systems. The trade-offs of each approach, selection criteria, and emerging trends for 2025 and beyond are discussed.

Keywords: Artificial Intelligence. Software Architecture. Microservices. MLOps. RAG.

RESUMEN
El rápido avance de la Inteligencia Artificial (IA) ha exigido que los equipos de ingeniería de software adopten arquitecturas robustas, escalables y observables para dar soporte a los modelos de aprendizaje automático en producción. Este trabajo presenta un análisis comparativo de las principales arquitecturas de software utilizadas actualmente en el desarrollo e implementación de sistemas basados en IA, abordando patrones como microservicios, arquitecturas sin servidor, pipelines MLOps, arquitectura orientada a eventos y sistemas RAG (Generación Aumentada por Recuperación). Se discuten las ventajas y desventajas de cada enfoque, los criterios de selección y las tendencias emergentes para 2025 y años posteriores.

Palabras clave: Inteligencia Artificial. Arquitectura de Software. Microservicios. MLOps. RAG.

1 INTRODUÇÃO

A integração de modelos de Inteligência Artificial (IA) em sistemas de produção deixou de ser uma tendência futurista e tornou-se uma realidade operacional para empresas de todos os portes. Contudo, a simples existência de um modelo de aprendizado de máquina bem treinado não garante sucesso em produção. A escolha da arquitetura de software que sustenta esse modelo é determinante para a confiabilidade, desempenho, escalabilidade e manutenabilidade do sistema como um todo (SCULLEY et al., 2015).

Ao contrário de sistemas tradicionais, aplicações baseadas em IA introduzem complexidades adicionais: os modelos degradam com o tempo (data drift), requerem pipelines de retreinamento, consomem recursos computacionais intensivos e frequentemente precisam de integração com fontes de dados heterogêneas em tempo real. Essas características tornam as escolhas arquiteturais ainda mais críticas.

Inseridos nesse contexto, encontram-se os engenheiros de software que precisam tomar decisões de design informadas. Este trabalho está dividido conforme a seguinte estrutura: na seção 2, são apresentados os tipos de arquitetura; a seção 3 detalha cada arquitetura individualmente; a seção 4 traz a análise comparativa; a seção 5 apresenta as tendências emergentes; e a seção 6 apresenta as conclusões e trabalhos futuros.

Além da análise conceitual, são consideradas as seguintes referências técnicas na elaboração deste trabalho:

Quadro 1: Principais referências utilizadas na elaboração deste artigo

AUTOR	TÍTULO	ANO
SCULLEY et al.	Hidden Technical Debt in ML Systems	2015
ZAHARIA et al.	Accelerating the ML Lifecycle with MLflow	2018
LEWIS et al.	Retrieval-Augmented Generation for NLP	2020
KREUZBERGER et al.	Machine Learning Operations (MLOps): Overview	2023
BAIR	The Shift from Models to Compound AI Systems	2024

Fonte: Elaborado pelo autor (2026).

2 TIPOS DE ARQUITETURA DE SOFTWARE PARA IA

Conforme Kreuzberger, Kühl e Hirschl (2023), os sistemas de IA em produção podem ser classificados quanto ao padrão arquitetural adotado em:

a) arquiteturas baseadas em microsserviços, nas quais cada modelo é encapsulado em um serviço independente;

b) arquiteturas serverless, orientadas à execução sob demanda sem gerenciamento de infraestrutura;

c) pipelines de MLOps, que automatizam o ciclo de vida completo do modelo de machine learning;

d) arquiteturas orientadas a eventos (EDA), que processam dados em fluxo contínuo em tempo real;

e) arquiteturas RAG (Retrieval-Augmented Generation), que combinam recuperação de informações com geração por LLMs.

3 ESTRUTURA DAS PRINCIPAIS ARQUITETURAS

3.1 ARQUITETURA DE MICROSSERVIÇOS COM IA EMBARCADA

A arquitetura de microsserviços decompõe o sistema em serviços independentes, cada um com responsabilidade bem definida. No contexto de IA, cada modelo pode ser encapsulado em seu próprio microsserviço, exposto via API REST ou gRPC. Ferramentas como Docker e Kubernetes são essenciais para orquestrar esses serviços em escala. O padrão Model-as-a-Service (MaaS) é uma extensão natural desse paradigma, sendo amplamente adotado por empresas como Netflix, Uber e Spotify (KREUZBERGER et al., 2023).

3.2 ARQUITETURA SERVERLESS PARA INFERÊNCIA

A abordagem serverless abstrai completamente a infraestrutura do desenvolvedor. No contexto de IA, é especialmente adequada para cargas de trabalho de inferência esporádica, como análise de imagens sob demanda ou classificação de documentos. O principal desafio reside no cold start: o tempo de inicialização de contêineres com modelos grandes pode ser inaceitável para aplicações sensíveis à latência. Segundo Zaharia et al. (2018), técnicas como provisioned concurrency têm mitigado esse problema.

3.3 PIPELINES DE MLOPS

MLOps representa a convergência de práticas de DevOps com o ciclo de vida de machine learning. Arquiteturalmente, MLOps se materializa em pipelines automatizados que integram coleta de dados, feature engineering, treinamento, avaliação, registro e implantação de modelos. Zaharia et al. (2018) descrevem essa abordagem como:

> "[...] uma plataforma unificada capaz de gerenciar o ciclo de vida completo de modelos de machine learning, desde a experimentação até a produção, garantindo rastreabilidade e reprodutibilidade em cada etapa do processo." (ZAHARIA et al., 2018, p. 14).

3.4 ARQUITETURA ORIENTADA A EVENTOS (EDA) COM IA

A arquitetura orientada a eventos é particularmente poderosa em sistemas de IA que precisam reagir a dados em tempo real. Tecnologias como Apache Kafka e AWS Kinesis servem como backbones de streaming, permitindo que modelos de ML consumam e processem eventos de forma assíncrona e escalável.

3.5 ARQUITETURA RAG (RETRIEVAL-AUGMENTED GENERATION)

A arquitetura RAG representa um dos padrões mais relevantes em 2024-2025 para sistemas baseados em Large Language Models (LLMs). Em vez de depender exclusivamente do conhecimento interno do modelo, o RAG combina recuperação de informações de bases de dados vetoriais com a capacidade generativa do LLM. Conforme Lewis et al. (2020), essa arquitetura endereça problemas críticos como alucinação e desatualização do conhecimento do modelo.

4 ANÁLISE COMPARATIVA

A escolha da arquitetura ideal depende de múltiplos fatores contextuais. A Tabela 1 apresenta uma comparação objetiva entre as abordagens discutidas neste trabalho:

Tabela 1: Comparação entre arquiteturas de software para sistemas de IA

Aspecto	Microsserviços	Serverless	MLOps	EDA/RAG
Escalabilidade	Alta	Alta	Média	Alta
Complexidade Ops	Alta	Média	Alta	Alta
Latência	Média	Alta*	Baixa	Média
Retreinamento	Manual	Manual	Automático	Contínuo
Ideal para	Modelos em prod.	Inferência esporádica	Governança de ML	Dados em tempo real

Fonte: Elaborado pelo autor (2026). (*) Cold start pode aumentar a latência.

Com base nos resultados observados na literatura, pôde-se identificar que cada arquitetura apresenta vantagens específicas para determinados cenários. Frente a isso, evidencia-se a necessidade de avaliação criteriosa do contexto antes de qualquer decisão de design.

5 TENDÊNCIAS EMERGENTES EM 2025

O ecossistema de arquiteturas para IA está em constante evolução. O Berkeley AI Research Institute (2024) identifica o surgimento dos chamados Compound AI Systems como a principal tendência arquitetural do período. Esses sistemas combinam múltiplos componentes de IA de forma orquestrada, substituindo progressivamente os modelos monolíticos.

Outra tendência relevante é a Edge AI, que consiste na execução de modelos diretamente em dispositivos de borda, como smartphones e sensores IoT, eliminando a dependência de servidores remotos e reduzindo a latência para valores próximos de zero. Frameworks como ONNX Runtime e TensorFlow Lite viabilizam essa abordagem.

Por fim, arquiteturas LLM-Native consolidam padrões como LLM Router, Model Cascade e Mixture of Experts (MoE) como referências de projeto reconhecidas pela comunidade de engenharia de software para 2025 e além.

6 CONCLUSÕES E TRABALHOS FUTUROS

Com base nos resultados analisados, observou-se que não existe uma arquitetura universal para sistemas de IA. A escolha ideal depende do contexto, dos requisitos de negócio e do estágio de maturidade do produto. Microsserviços oferecem flexibilidade para modelos em produção; serverless é adequado para inferências esporádicas; MLOps garante governança e automação; EDA habilita IA em tempo real; e RAG potencializa LLMs com conhecimento atualizado.

Identificou-se, inclusive, que o denominador comum entre todas as arquiteturas bem-sucedidas é a observabilidade: sistemas de IA em produção exigem monitoramento contínuo de métricas de infraestrutura e de modelo. As escolhas arquiteturais feitas hoje determinarão a capacidade das organizações de escalar e evoluir seus sistemas de IA nos próximos anos.

Como oportunidade para trabalhos futuros, planeja-se realizar um estudo de caso aplicando as arquiteturas analisadas em um projeto real desenvolvido na Faculdade FUCAPI, avaliando métricas objetivas de desempenho, escalabilidade e custo operacional. Pretende-se também expandir a análise para arquiteturas de sistemas multi-agentes com LLMs.

REFERÊNCIAS

BERKELEY AI RESEARCH INSTITUTE. The shift from models to compound AI systems. Disponível em: <https://bair.berkeley.edu>. Acesso em: 27 maio 2026.

KREUZBERGER, Dominik; KÜHL, Niklas; HIRSCHL, Sebastian. Machine learning operations (MLOps): overview, definition, and architecture. IEEE Access, v. 11, p. 31866-31879, 2023.

LAKATOS, Eva Maria; MARCONI, Marina de Andrade. Fundamentos de metodologia científica. 3. ed. rev. e ampl. São Paulo: Atlas, 1991. 270 p.

LEWIS, Patrick et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, 33., 2020, [S. l.]. Anais [...]. [S. l.]: NeurIPS, 2020. p. 9459-9474.

SCULLEY, D. et al. Hidden technical debt in machine learning systems. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, 28., 2015. Anais [...]. [S. l.]: NeurIPS, 2015.

SUTHERLAND, Jeff. Scrum: a arte de fazer o dobro do trabalho na metade do tempo. São Paulo: LeYa, 2014.

ZAHARIA, Matei et al. Accelerating the machine learning lifecycle with MLflow. IEEE Data Engineering Bulletin, v. 41, n. 4, p. 39-45, 2018.