



IA APLICADA

Mapa de IA 2026

Antes de construir qualquer coisa
um guia prático para entender
sistemas, agentes, risco e produção.

por Cesar Augusto

EDIÇÃO 000 · v2026.1

Handbook em português sobre projetos de IA

Antes do modelo, o mapa.

IA virou uma palavra grande demais. Em 2026, ela aparece em produto, curso, vaga, pitch, reunião e tutorial. O problema é que muita coisa diferente está sendo chamada pelo mesmo nome.

Este mapa organiza o básico antes da construção: o que é IA, como a pilha funciona, o que LLMs fazem, que tipos de sistema existem, onde agentes entram, e por que dados, avaliação, custo e risco não são detalhes finais.

A ideia não é decorar termos nem escolher uma marca favorita. É criar vocabulário, critério e uma trilha prática para sair do hype e começar a construir com responsabilidade.

Como usar este handbook

Leia como um mapa de decisão: primeiro nomeie o sistema, depois escolha a arquitetura, defina dados, avalie risco e só então pense em escala.



Índice

MENU DO MAPA

01

O que é IA, de verdade

02

A pilha da IA em 2026

03

LLMs: o que fazem e o que não fazem

04

Os 6 tipos de sistema que você vai encontrar

05

Agentes, ferramentas e o que mudou em 2024-2026

06

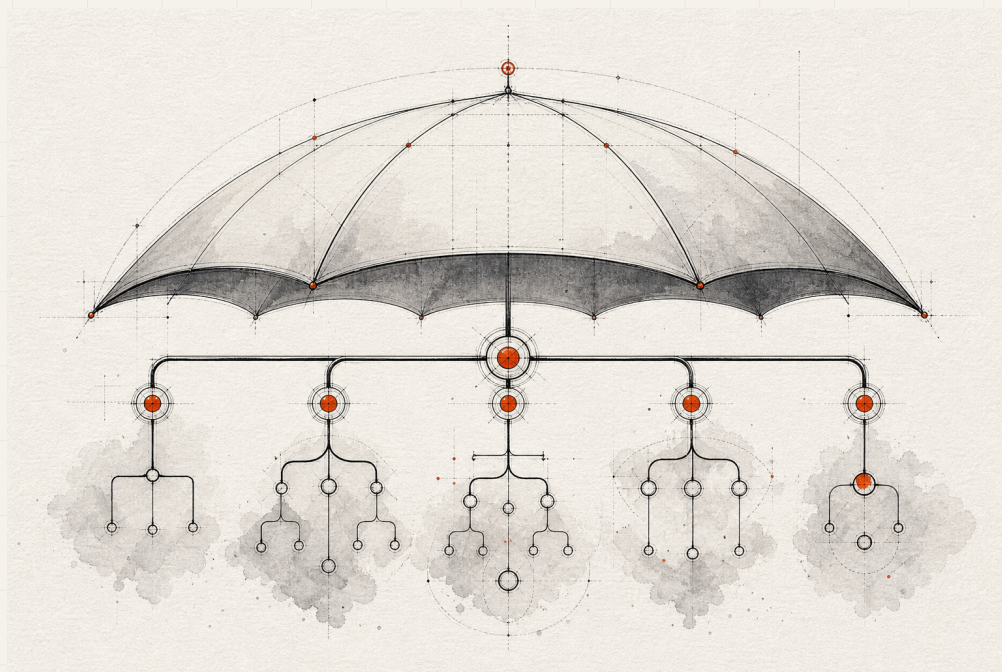
Dados, avaliação, custo e risco

07

Como estudar IA em português



O que é IA, de verdade



IA não é uma coisa só. É um guarda-chuva que inclui automação inteligente, aprendizado de máquina, IA generativa, modelos de linguagem e agentes. Entender os nomes evita comprar marketing como se fosse engenharia.

3 perguntas para desmascarar 'nossa IA'

- Qual modelo ou técnica está por trás?
- Com quais dados foi treinado ou conectado?
- Como vocês medem se a resposta está certa?

IA pensa como um humano

- Modelos atuais encontram padrões e geram saídas prováveis. Parecem entender porque linguagem é familiar, mas não há consciência, intenção nem experiência de mundo.



Definição sem marketing

Inteligência artificial é o nome dado a sistemas que executam tarefas que parecem inteligentes: responder perguntas, reconhecer imagens, prever resultados, gerar textos ou apoiar decisões.

Isso não significa que a máquina pensa, sente ou entende como uma pessoa. Significa que ela produz comportamento útil a partir de regras, dados, modelos ou geração probabilística.

Automação não é sempre IA

Um fluxo que aprova pedidos abaixo de R\$ 500 segue regras fixas. Isso pode ser útil, mas não é necessariamente IA.

Um sistema que aprende padrões em dados ou gera respostas novas a partir de contexto trabalha de outro jeito: ele se adapta ao que recebe, dentro dos limites do modelo.

Três eras para entender o mercado

Era	Período	Ideia central	Exemplo
IA simbólica	1950-1990	Regras e lógica escritas por humanos	Xadrez, sistemas especialistas
Aprendizado de máquina	1990-2017	Modelos aprendem padrões em dados	Filtro de spam, recomendação
IA generativa	2017-hoje	Modelos grandes criam conteúdo	ChatGPT, Claude, Gemini

Termos essenciais

- IA: guarda-chuva para sistemas com comportamento inteligente.
- Machine learning: IA que aprende padrões a partir de dados.
- Deep learning: machine learning com redes neurais profundas.
- IA generativa: cria texto, imagem, áudio, vídeo ou código.
- LLM: modelo de linguagem grande, usado em chatbots e copilotos.
- Agente: sistema que planeja passos, chama ferramentas e tenta concluir um objetivo.

O que vendem como IA, mas não é

- Fluxos rígidos com if/else.
- Chatbots de árvore de decisão.
- Botão novo com integração simples a API externa.
- Autocomplete reembalado como produto de IA.



A pilha da IA em 2026



Projeto de IA que começa pelo modelo costuma quebrar cedo. Todo sistema sério precisa de problema claro, dados confiáveis, modelo adequado, orquestração e operação.

Checklist antes de prototipar

- Problema escrito em uma frase.
- Lista dos dados disponíveis e do que falta.
- Modelo candidato e plano B.

Escolher modelo antes do problema

- Dizer 'vamos usar o modelo X porque é melhor' não basta. Melhor em quê? Com quais dados? Medido como? Sem resposta, é compra de religião, não construção de sistema.



As 5 camadas

Camada	O que é	Pergunta-chave
Problema	A tarefa real a resolver	Qual trabalho melhora de verdade?
Dados	Fonte, qualidade e permissão	De onde vem a informação?
Modelo	Motor que processa ou gera	Qual técnica cabe aqui?
Orquestração	Prompt, RAG, ferramentas e agentes	Como o modelo se conecta ao mundo?
Operação	Logs, avaliações, custo e correção	Como sabemos se continua funcionando?

Infraestrutura

Antes de falar de chatbot, existe hardware, rede, nuvem, custo, latência e decisão sobre onde o dado trafega.

APIs em nuvem aceleram protótipos. Servidores locais ou VPCs dão mais controle, mas exigem equipe e operação. Arquiteturas híbridas são comuns quando há dados sensíveis.

Ecossistema de modelos

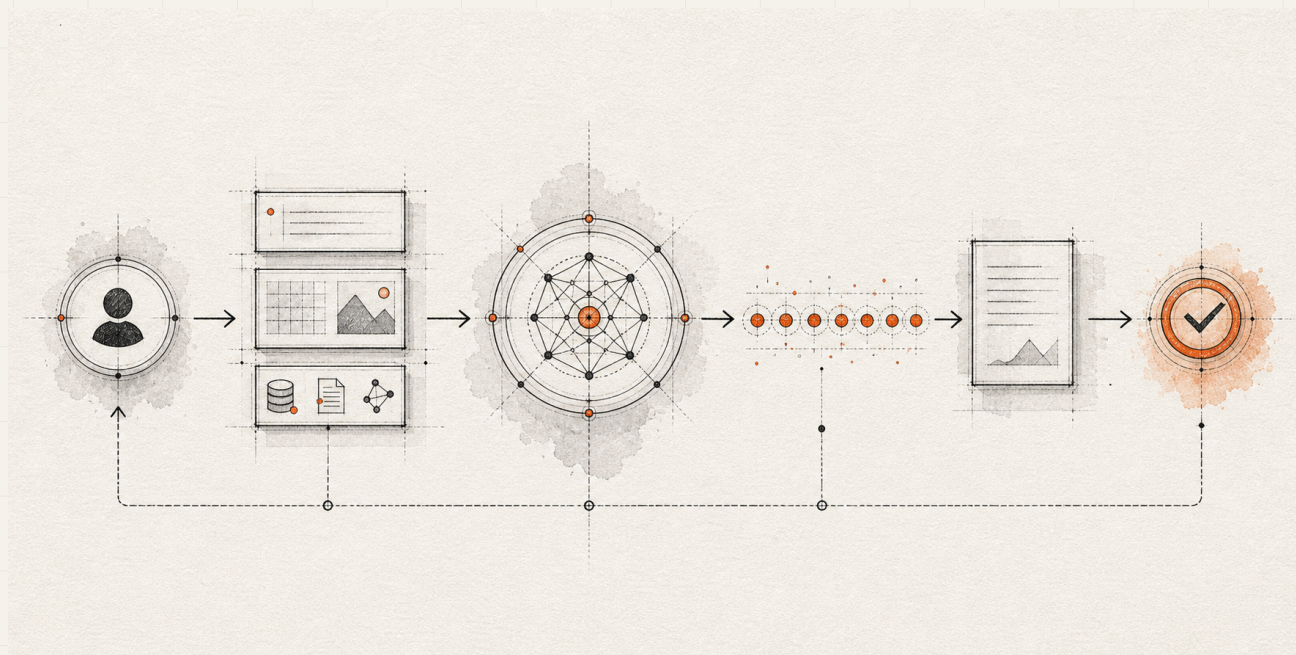
- Modelos fechados via API entregam velocidade e alta performance, mas exigem atenção a dados e contrato.
- Modelos pesos abertos entregam controle e customização, mas exigem infraestrutura e MLOps.
- Modelos com foco em português podem render melhor em tarefas locais, mas precisam ser avaliados no caso real.
- Em 2026, a escolha deve ser feita por tarefa, custo, risco e política de dados, não por torcida de marca.

Ferramentas e protocolos

- Orquestração: LangChain, LangGraph, CrewAI, Google ADK.
- Bancos vetoriais: Pinecone, Weaviate, Chroma, pgvector.
- Observabilidade: Langfuse, LangSmith, Phoenix.
- MCP: protocolo para conectar modelos a ferramentas e fontes de dados de forma padronizada.



LLMs: o que fazem e o que não fazem



LLMs são poderosos para linguagem, estruturação e código com supervisão. Mas não são oráculos: podem alucinar, errar fatos atuais, falhar em matemática e variar entre execuções.

Antes de confiar na resposta

- Se precisa de fato verificável, exija fonte ou RAG.
- Se envolve número ou lógica, valide com ferramenta externa.
- Se vai para cliente ou decisão crítica, inclua revisão humana.

O modelo entendeu minha pergunta

- Ele gerou uma sequência compatível com padrões aprendidos. Isso parece entendimento, mas o mecanismo é predição, não compreensão.



Como um LLM funciona

Antes de conversar com você, o modelo foi exposto a grandes volumes de texto. Ele não leu como humano: aprendeu padrões estatísticos de linguagem.

Cada resposta é construída token por token. O modelo estima qual pedaço de texto provavelmente vem a seguir, dado o contexto.

Conceitos essenciais

Conceito	O que é	Por que importa
Prompt	Instrução e contexto enviados ao modelo	Define tom, formato e qualidade
Token	Unidade mínima de texto processada	Define custo, limite e velocidade
Janela de contexto	Quanto texto o modelo enxerga de uma vez	Limita histórico e documentos longos
Temperatura	Controle de aleatoriedade	Baixa para consistência; alta para criatividade
Data de corte	Limite do conhecimento treinado	Pode deixar o modelo desatualizado
Alucinação	Resposta confiante e factualmente errada	Risco central em produção

O que LLMs fazem bem

- Resumir, reescrever, traduzir e estruturar texto.
- Gerar código com revisão humana.
- Classificar intenção e extrair campos de texto.
- Ajudar em brainstorming e planejamento delimitado.



O que LLMs fazem mal

- Fatos atuais sem busca, RAG ou fonte externa.
- Matemática e lógica complexa sem verificador.
- Acesso a CRM, ERP ou banco de dados sem integração.
- Garantir respostas idênticas em execuções diferentes.
- Compreender no sentido humano.

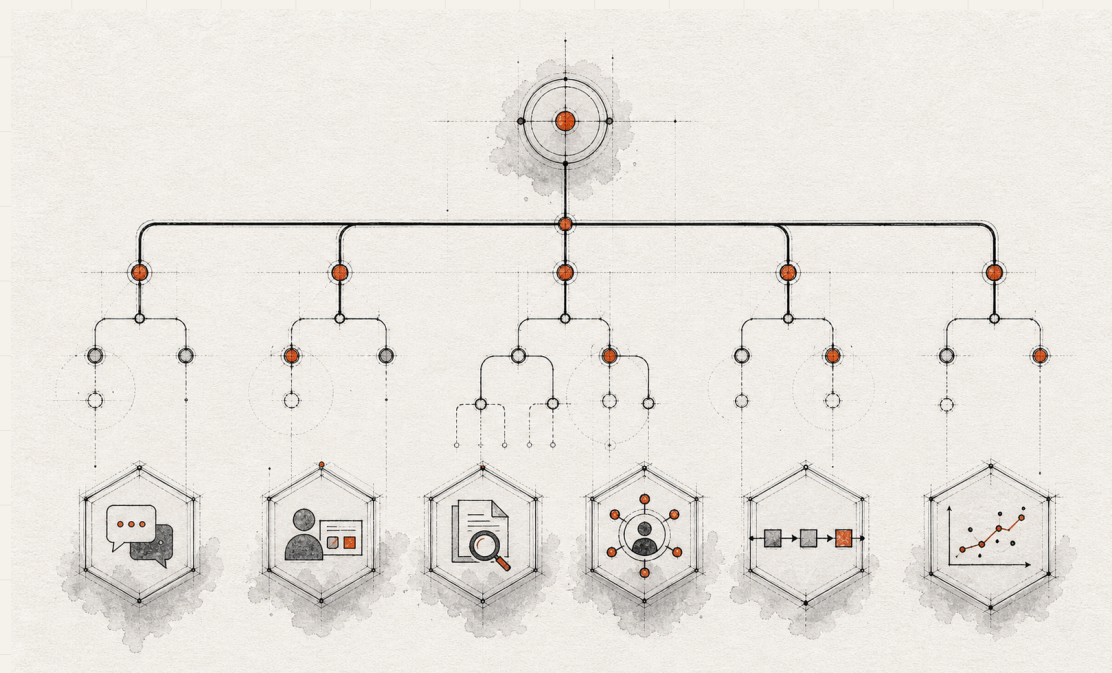
Multimodalidade

Modelos modernos processam texto, imagem, áudio e vídeo. Isso permite análise de documentos escaneados, acessibilidade, transcrição, inspeção visual e novos fluxos de trabalho.

A mesma capacidade amplia riscos: deepfakes, documentos falsos, evidências manipuladas e jailbreaks escondidos em imagem.



Os 6 tipos de sistema que você vai encontrar



Nem tudo precisa virar agente. Em 2026, os padrões mais úteis são chatbot simples, copiloto, RAG, agente com ferramentas, pipeline automatizado e ML clássico.

Meu caso é qual tipo?

- Escrever melhor: copiloto.
- Perguntar sobre PDFs: RAG.
- Fazer sozinho em 5 sistemas: agente, com cuidado.

Agente autônomo como primeiro projeto

- Demo impressiona. Produção cobra. Sem avaliação, limite de passos, log e revisão humana em ações críticas, agente vira risco operacional.



Os 6 tipos

Tipo	Quando usar	Risco principal
Chatbot simples	FAQ genérico, brainstorming, protótipo rápido	Alucinar ou estar desatualizado
Copiloto	Quando humano revisa e decide	Baixo, se a revisão for real
RAG	Perguntas sobre documentos internos	Buscar trecho errado ou base desatualizada
Agente com ferramentas	Automação multi-passo com ações	Loop, custo e ação irreversível
Pipeline automatizado	Volume alto com etapas fixas	Erro em escala
ML clássico	Previsão, classificação e anomalia em dados estruturados	Modelo ruim ou dados enviesados

Como escolher

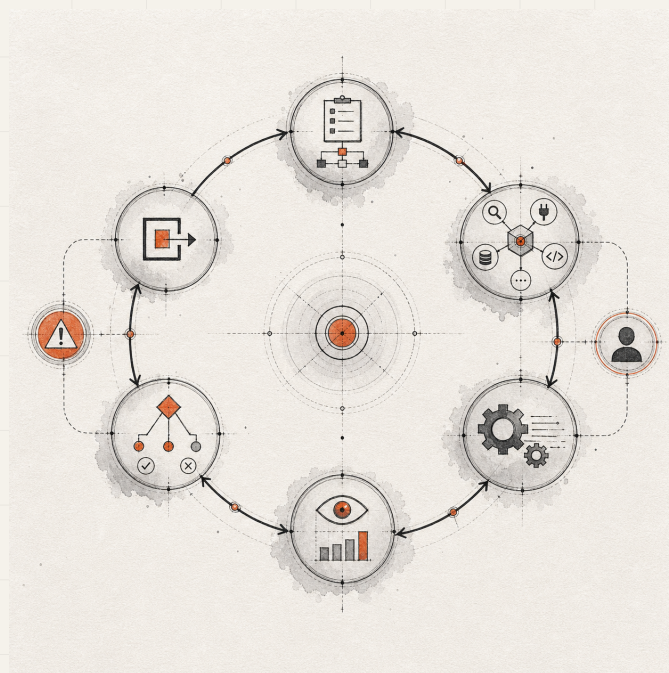
- Precisa citar documentos internos? Use RAG.
- Precisa executar ações em sistemas? Use agente com limites.
- Precisa prever número ou classificar dado tabular? Use ML clássico.
- Humano já revisa naturalmente? Use copiloto.
- Volume alto e etapas fixas? Use pipeline.
- Nada disso? Comece com chatbot simples, com cautela.

Tabela comparativa

Tipo	Complexidade	Custo	Risco	MVP
Chatbot	Baixa	Baixo	Médio	Dias
Copiloto	Baixa	Baixo	Baixo	Dias
RAG	Média	Médio	Médio	Semanas
Agente	Alta	Alto	Alto	Semanas a meses
Pipeline	Média	Médio	Médio	Semanas
ML clássico	Alta	Variável	Baixo a médio	Meses



Agentes, ferramentas e o que mudou em 2024-2026



Agente é LLM em loop: recebe objetivo, planeja, escolhe ferramenta, executa, observa e decide o próximo passo. É poderoso, mas transforma erro textual em ação errada se não houver limites.

Mínimo viável de um agente seguro

- Objetivo escrito em uma frase.
- Catálogo mínimo de ferramentas.
- Limite de passos e tempo limite.

Agente sem limite de passos

- Sem teto, o loop pode rodar indefinidamente, consumir tokens, chamar APIs e produzir lixo. Defina limite de passos, tempo limite e orçamento por execução.



De chatbot a agente

Chatbot: uma pergunta, uma resposta.

Agente: um objetivo, vários passos, ferramentas, observação e decisão até concluir, falhar ou pedir ajuda.

Anatomia de um agente

- Objetivo claro.
- Decomposição da tarefa.
- Seleção de ferramenta.
- Execução e observação.
- Loop com parada, limite ou escalação humana.

Ferramentas

- Busca: web, documentos internos e banco vetorial.
- API: CRM, ERP, calendário e pagamentos.
- Código: calcular, transformar ou validar dados.
- Banco de dados: consultas com permissão controlada.
- Comunicação: e-mail e Slack, sempre com aprovação em ações críticas.



MCP

O Model Context Protocol padroniza como aplicações expõem ferramentas e contexto para modelos. Ele reduz integrações artesanais e aumenta portabilidade, mas não substitui segurança, avaliação, logs e governança.

Multi-agente

Sistemas multi-agente fazem sentido quando há etapas especializadas, permissões distintas ou contexto grande demais para um agente único. Para tarefas pequenas, multi-agente pode ser teatro caro: aumenta latência, custo e pontos de falha.

O que mudou

Aspecto	2022	2026
Foco	ChatGPT como novidade	IA embutida em produtos
Engenharia	Prompt crafting	Sistemas, avaliações e operação
Dados	Opcional	RAG quase obrigatório
Agente	Demo em vídeo	Risco operacional real
Modalidade	Só texto	Multimodal como padrão



Dados, avaliação, custo e risco



Dados, eval, custo e risco não são fase 2. São parte do MVP. Se o modelo errar amanhã de manhã, alguém precisa perceber, reproduzir e corrigir.

Se errar amanhã, quem percebe?

- Alerta automático de erro, avaliação, latência ou custo.
- Responsável humano definido.
- Procedimento de rollback.

Escalar sem eval

- Abrir para 10 mil usuários sem conjunto de referência é loteria. Você só descobre os buracos quando viram print, reclamação ou incidente.



Dados

- Qualidade vence volume: 50 documentos limpos podem superar mil PDFs desorganizados.
- LGPD exige base legal, finalidade, minimização e anonimização quando possível.
- Dados sensíveis no prompt exigem política clara, especialmente ao usar API de terceiro.
- Dados sintéticos ajudam em teste, mas não devem substituir realidade sem validação.
- Viés nos dados vira viés no sistema.

Avaliação

Parecer bom em três testes não basta. Eval é o antídoto contra achismo.

Use avaliação humana para qualidade especializada e avaliação automática para velocidade. Em produção, combine as duas.

- Golden set: conjunto fixo de perguntas e respostas esperadas.
- Teste de regressão: rode o conjunto de referência ao trocar modelo, prompt ou base RAG.
- Métricas: aceite do usuário, escalação humana, latência, custo e alucinações detectadas.

Custo real

- APIs cobram por tokens de entrada e saída.
- Agentes multi-passo multiplicam custo a cada iteração.
- RAG adiciona embedding, busca vetorial, reranking e armazenamento.
- Observabilidade, retries e manutenção também custam.

Escala	Requisições por dia	Perfil de custo
Pessoal	Menos de 100	Quase irrelevante em API
Time pequeno	100 a 1.000	Monitorar todo mês
Produto	1.000 a 50.000	Otimização obrigatória
Escala	50.000+	Modelo menor, cache e processamento em lote



Riscos principais

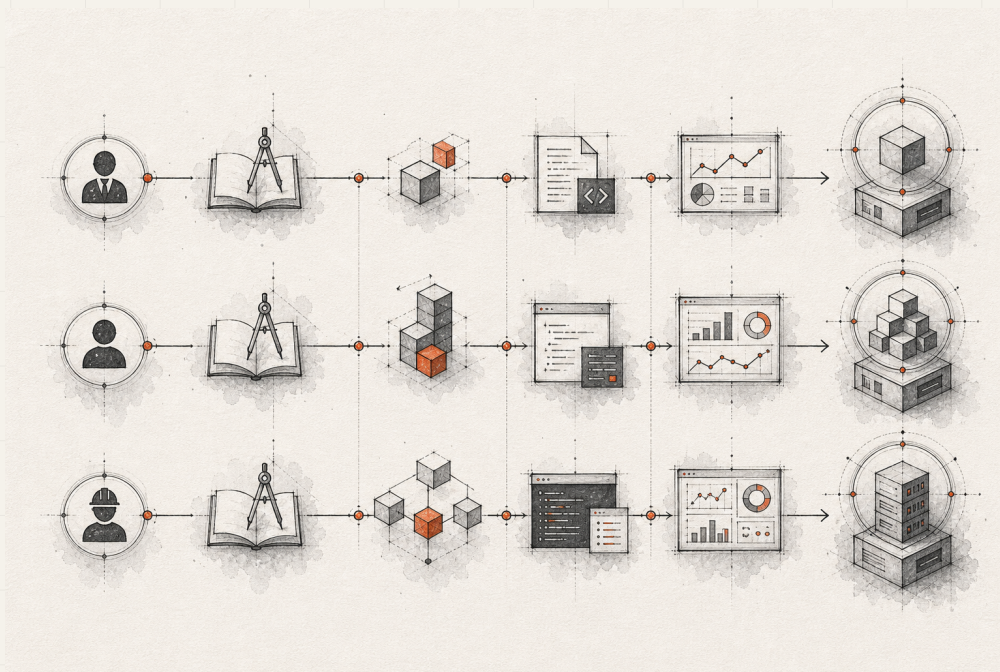
Risco	Mitigação mínima
Injeção de prompt	Separar instrução de dado e validar entrada
Vazamento sensível	Política de dados, anonimização e contrato adequado
Fonte não confiável	Allowlist e validação de documentos
Alucinação	RAG, citação e revisão humana
Ação não autorizada	Permissões, aprovação e limite de ferramentas
Cadeia de suprimentos	Auditar libs, modelos e APIs

Governança mínima

- Definir quem responde quando a IA erra.
- Definir o que pode entrar no sistema.
- Guardar prompt, resposta, ferramentas chamadas e timestamp.
- Escalar para humano em baixa confiança, fora de escopo, ação irreversível ou pedido explícito do usuário.



Como estudar IA em português



Depois do mapa, o próximo passo não é mais teoria infinita. É escolher uma trilha, construir algo pequeno, medir e documentar o aprendizado.

Seus três compromissos

- Um projeto real em 30 dias.
- Golden set antes de mostrar para outras pessoas.
- Um post ou nota documentando o que aprendeu.



Antes de escolher um curso

- Defina seu perfil: curioso, construtor iniciante ou engenheiro.
- Defina seu objetivo: usar, prototipar ou operar.
- Seja honesto sobre tempo disponível: 2 horas por semana para literacia, 5 a 10 horas para protótipo, 15 a 20+ horas para produção.

Trilhas por perfil

Perfil	Tempo por semana	Próximo passo
Curioso ou gestor	2 a 4 horas	Elements of AI + posts Korvo
Construtor iniciante	5 a 10 horas	RAG com PDFs + CS50 AI
Engenheiro	15+ horas	Agente com avaliação + observabilidade

O que pular em 2026

- Cursos de 2 horas que prometem domínio completo.
- Tutoriais só de prompt sem sistema.
- Pitch de 'substituir o time com agentes'.
- Material antigo com modelo, preço ou API descontinuados.



O que vale ler em inglês

- Documentação oficial do modelo que você usa.
- OWASP Top 10 for LLM Applications.
- Papers com notas comentadas e fichamentos em português.

Projeto de 7 dias pós-mapa

Dia	Tarefa
1	Escolher um problema real do trabalho ou estudo
2	Definir tipo de sistema: copiloto, RAG ou pipeline
3	Montar protótipo mínimo
4	Criar conjunto de referência com 10 perguntas
5	Rodar avaliação e registrar falhas
6	Ajustar prompt, dados ou tipo de sistema
7	Documentar aprendizado em post, nota ou README



Agora construa uma coisa pequena.

Você agora tem um mapa mental completo: o que é IA, como a pilha funciona, o que LLMs fazem, que tipo de sistema montar, como pensar agentes e como medir risco, custo e qualidade.

O próximo passo não é escolher o curso perfeito. É construir uma coisa pequena, medir, errar em privado e melhorar em público.

Comece com um problema real. Faça caber em uma semana. Meça antes de escalar.

Regra de saída

Faça caber em uma semana. Meça antes de mostrar. Documente o aprendizado.





Korvo.dev

Ideias práticas sobre IA aplicada.