

E-BOOK · OSNEY SOUZA

IA que não inventa número

Como levar IA para documentos e números sem alucinação

AMOSTRA GRÁTIS

Osney A. de Souza

AI Engineer · osneyandrade.github.io

IA que não inventa número

Como levar IA para documentos e números sem alucinação

Osney A. de Souza · AI Engineer

Edição: outubro 2026

© 2026 Osney A. de Souza. Todos os direitos reservados.

Licença de uso individual: não redistribua este arquivo. Para uso em equipe ou empresa, peça uma licença.

Os exemplos usam dados fictícios. Este material não contém código, dados ou telas de clientes nem de empregadores do autor.

<https://osneyandrade.github.io>
juniorthesouza017@gmail.com

Sumário

1	A fronteira entre leitura e cálculo	5
	Entender por que modelos de linguagem não podem gerar valores financeiros e como estruturar um fluxo onde a IA lê e o código calcula.	
2	Leitura de documentos não estruturados	<i>Continua na versão completa</i>
	Implementar extração de dados de PDFs escaneados, fotos e planilhas usando OCR e modelos de linguagem para localizar campos sem inventar conteúdo.	
3	Oráculo determinístico e gate de sigilo	<i>Continua na versão completa</i>
	Criar uma camada de validação que recálcula valores e veta mapeamentos errados antes que o dado entre no sistema de destino.	
4	RAG com citação e embeddings locais	<i>Continua na versão completa</i>
	Construir sistemas de busca em metadados e documentos usando embeddings e modelos locais para evitar vazamento de dados sensíveis.	
5	Agentes com tool-calling para conferência	<i>Continua na versão completa</i>
	Desenvolver agentes que chamam ferramentas para verificar datas, cadastros e totais, delegando a lógica de negócio ao código.	
6	Cascata de custos e avaliação contínua	<i>Continua na versão completa</i>
	Organizar o uso de modelos em nuvem e locais por custo e estabelecer métricas de erro para monitoramento em produção.	
7	Integração com sistemas de destino	<i>Continua na versão completa</i>
	Conectar o pipeline de leitura e validação ao sistema de gestão, gerando painéis de exceções para revisão humana.	

Antes de começar

Para quem é

- Desenvolvedores, líderes técnicos e analistas que precisam usar IA com documentos e números (financeiro, fiscal, operações)

O que você vai conseguir fazer

- Entender por que modelos de linguagem não podem gerar valores financeiros e como estruturar um fluxo onde a IA lê e o código calcula.
- Implementar extração de dados de PDFs escaneados, fotos e planilhas usando OCR e modelos de linguagem para localizar campos sem inventar conteúdo.
- Criar uma camada de validação que recalcula valores e veta mapeamentos errados antes que o dado entre no sistema de destino.
- Construir sistemas de busca em metadados e documentos usando embeddings e modelos locais para evitar vazamento de dados sensíveis.
- Desenvolver agentes que chamam ferramentas para verificar datas, cadastros e totais, delegando a lógica de negócio ao código.
- Organizar o uso de modelos em nuvem e locais por custo e estabelecer métricas de erro para monitoramento em produção.
- Conectar o pipeline de leitura e validação ao sistema de gestão, gerando painéis de exceções para revisão humana.

CAPÍTULO 1

A fronteira entre leitura e cálculo

OBJETIVO *Entender por que modelos de linguagem não podem gerar valores financeiros e como estruturar um fluxo onde a IA lê e o código calcula.*

O perigo da geração livre de valores

Modelos de linguagem são treinados para prever a próxima palavra em uma sequência. Eles aprendem padrões linguísticos, não matemática exata. Quando um modelo recebe um prompt como "qual é o total da nota fiscal?", ele tenta estimar o valor com base no que leu no texto. Isso gera alucinações numéricas. O modelo pode inventar um número que não existe no documento original, ou arredondar de forma inconsistente. Em auditoria, contabilidade e operações financeiras, um erro de digitação ou arredondamento não é um detalhe; é uma falha crítica que pode levar a multas, perda de confiança e prejuízo direto.

A fronteira fundamental em sistemas de IA para documentos e números é clara: a IA lê, o código calcula. A IA é excelente para extrair texto de um PDF escaneado, identificar que um campo é "Valor Total" e entender o contexto da frase. No entanto, a IA nunca deve ser a fonte primária de um valor monetário ou de um cálculo lógico. O valor final deve sempre vir de uma leitura estruturada ou de um recálculo feito por código determinístico.

Como estruturar o fluxo de leitura e validação

Para construir um sistema confiável, você deve separar as responsabilidades. O fluxo ideal começa com a extração de texto não estruturado. A IA lê o documento e identifica os campos. Em seguida, o sistema extrai os valores numéricos brutos. Esses valores brutos são passados para uma camada de código que realiza a validação e o cálculo.

Considere o seguinte cenário hipotético:

exemplo Uma nota fiscal digitalizada contém três linhas de itens:

- Item A: Quantidade 10, Preço Unitário 100,00
- Item B: Quantidade 5, Preço Unitário 200,00
- Item C: Quantidade 2, Preço Unitário 150,00

O documento também possui um campo "Subtotal" preenchido manualmente como 2.000,00.

Um modelo de linguagem poderia ler o documento e dizer: "O subtotal é 2.000,00". Isso parece correto, mas se houver um erro de digitação no documento original (por exemplo, o subtotal escrito como 2.005,00), o modelo pode aceitar o valor errado se ele não for treinado especificamente para detectar inconsistências.

O código determinístico, por outro lado, lê as linhas de itens, multiplica a quantidade pelo preço unitário de cada um e soma os resultados. O resultado do código é 3.500,00. O sistema compara esse resultado calculado com o subtotal do documento. Se houver diferença, o sistema gera uma exceção para revisão humana. O valor final aceito pelo sistema é o calculado pelo código, não o lido pelo modelo.

Implementação técnica com Python

Abaixo, um trecho de código Python que ilustra essa separação. O script lê os dados extraídos (simulados) e calcula o total. Ele não usa IA para gerar o valor, mas sim para validar se o valor extraído pelo OCR ou IA está coerente com o cálculo.

```

import pandas as pd

# Simulação de dados extraídos de um PDF ou OCR
# O modelo de linguagem forneceu os campos, mas não o valor final
extracted_data = {
    "item_a_qty": 10,
    "item_a_price": 100.00,
    "item_b_qty": 5,
    "item_b_price": 200.00,
    "item_c_qty": 2,
    "item_c_price": 150.00,
    "document_subtotal": 2000.00 # Valor lido no documento, possivelmente
    errado
}

# Função determinística de cálculo
def calculate_total(data):
    total = (data["item_a_qty"] * data["item_a_price"]) + \
            (data["item_b_qty"] * data["item_b_price"]) + \
            (data["item_c_qty"] * data["item_c_price"])
    return total

# Cálculo do valor real
calculated_total = calculate_total(extracted_data)

# Validação: O valor calculado deve ser igual ao valor do documento
if abs(calculated_total - extracted_data["document_subtotal"]) > 0.01:
    print(f"ALERTA: Discrepância detectada.")
    print(f"Valor calculado: {calculated_total}")
    print(f"Valor no documento: {extracted_data['document_subtotal']}")
    # Aqui acionaria um gate de exceção para revisão humana
else:
    print(f"Validação aprovada. Total: {calculated_total}")

```

Neste exemplo, o código ignora o valor do documento se ele não corresponder ao cálculo. O sistema opera com base na verdade matemática, não na percepção do modelo.

Checklist de implementação

Para garantir que seu sistema não invente números, siga estas diretrizes:

- ☐ Nunca use o output de um LLM como fonte primária de valores monetários.
- ☐ Sempre extraia os valores brutos do documento antes de processá-los.
- ☐ Implemente uma função de cálculo determinístico em código para validar os valores.
- ☐ Compare o valor calculado com o valor extraído do documento.
- ☐ Gere uma exceção automática se a diferença for maior que uma tolerância mínima (ex: 0,01).
- ☐ Use um oráculo determinístico para vetar mapeamentos errados antes de salvar.
- ☐ Mantenha os modelos de linguagem localizados para dados sensíveis, mas use código para a lógica financeira.

Próximos passos

Agora que entendemos que a IA lê e o código calcula, precisamos abordar como lidar com documentos que não são estruturados, como PDFs escaneados ou fotos. A leitura desses documentos exige técnicas específicas para garantir que o texto seja extraído com precisão antes de ser processado.

Sobre o autor



Osney A. de Souza é AI Engineer em Joinville (SC), com 5 anos de desenvolvimento de software. Hoje constrói e opera em produção uma plataforma de auditoria assistida por LLM: leitura de documentos não estruturados (PDF escaneado, Excel, OFX, CSV), RAG com embeddings e pgvector, agentes com tool-calling, MCP, Structured Outputs e modelos locais e em nuvem em cascata por custo, com cerca de 3.100 testes automatizados. A regra central do trabalho dele: o modelo nunca produz o número final. Antes, trabalhou com visão computacional e leitura de placas em escala, back-end, aplicativos móveis e integração de sistemas. Cursa Engenharia de Software na Univille.

Site osneyandrade.github.io
GitHub github.com/OsneyAndrade
Contato juniorthesouza017@gmail.com

AMOSTRA GRÁTIS

Gostou do primeiro capítulo?

A versão completa traz também:

- Leitura de documentos não estruturados
- Oráculo determinístico e gate de sigilo
- RAG com citação e embeddings locais
- Agentes com tool-calling para conferência
- Cascata de custos e avaliação contínua
- Integração com sistemas de destino

Baixar o e-book agora

<https://osneyandrade.github.io/ia-que-nao-inventa-numero/>