IA Ágil

PDLC/SDLC com IA

Mauro Widman - AI Coach

Cronograma

1. Fundamentos LLM, Contexto e Desenvolvimento Usando IA
2. Projeto Greenfield - Desenvolvendo um projeto do zero
3. Projeto Brownfield - Desenvolvendo funcionalidades em um projeto
4. Refatoração - Atualizando um projeto legado
5. Criando agentes, tarefas e templates especializados

aegro

IA Ágil

PDLC/SDLC com IA

Dia 1 - Fundamentos de LLM, Contexto e Desenvolvimento Usando IA

aegro

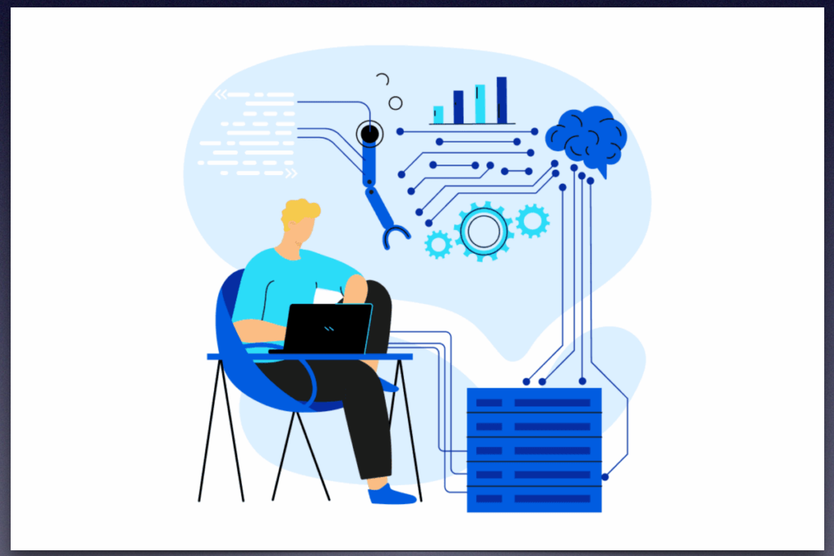
Agenda do Dia 1

- Entendendo LLM (Large Language Models)
- Técnicas eficazes de engenharia de prompt
- Aprendendo engenharia de contexto para otimizar respostas de IA
- Praticando geração de código utilizando IA
- Ética e limites no desenvolvimento com IA

aegro

Introdução

Por que utilizar IA para o ciclo de desenvolvimento?



- IA não é mais opcional - **é essencial**
- A metodologia de **PDLC/SDLC** conecta a **área de negócios** com o **desenvolvimento** usando **IA**

aegro

Introdução ao LLM

- Redes neurais são treinadas em grandes quantidades de texto e informação
- Aprendem padrões, relacionamentos e estruturas de linguagem
- Podem entender, gerar e manipular texto
- Fundação das ferramentas de IA modernas de desenvolvimento

aegro

Como os LLMs funcionam?

Texto de Entrada (Prompt) → Tokenização → Embedding → Camadas de Atenção → Saída

TOKENS

Unidades básicas de texto
(Palavras, sub-palavras, caracteres)

EMBEDDINGS

Representações numéricas dos tokens

CAMADAS DE ATENÇÃO

Ponderam a importância de diferentes tokens

JANELA DE CONTEXTO

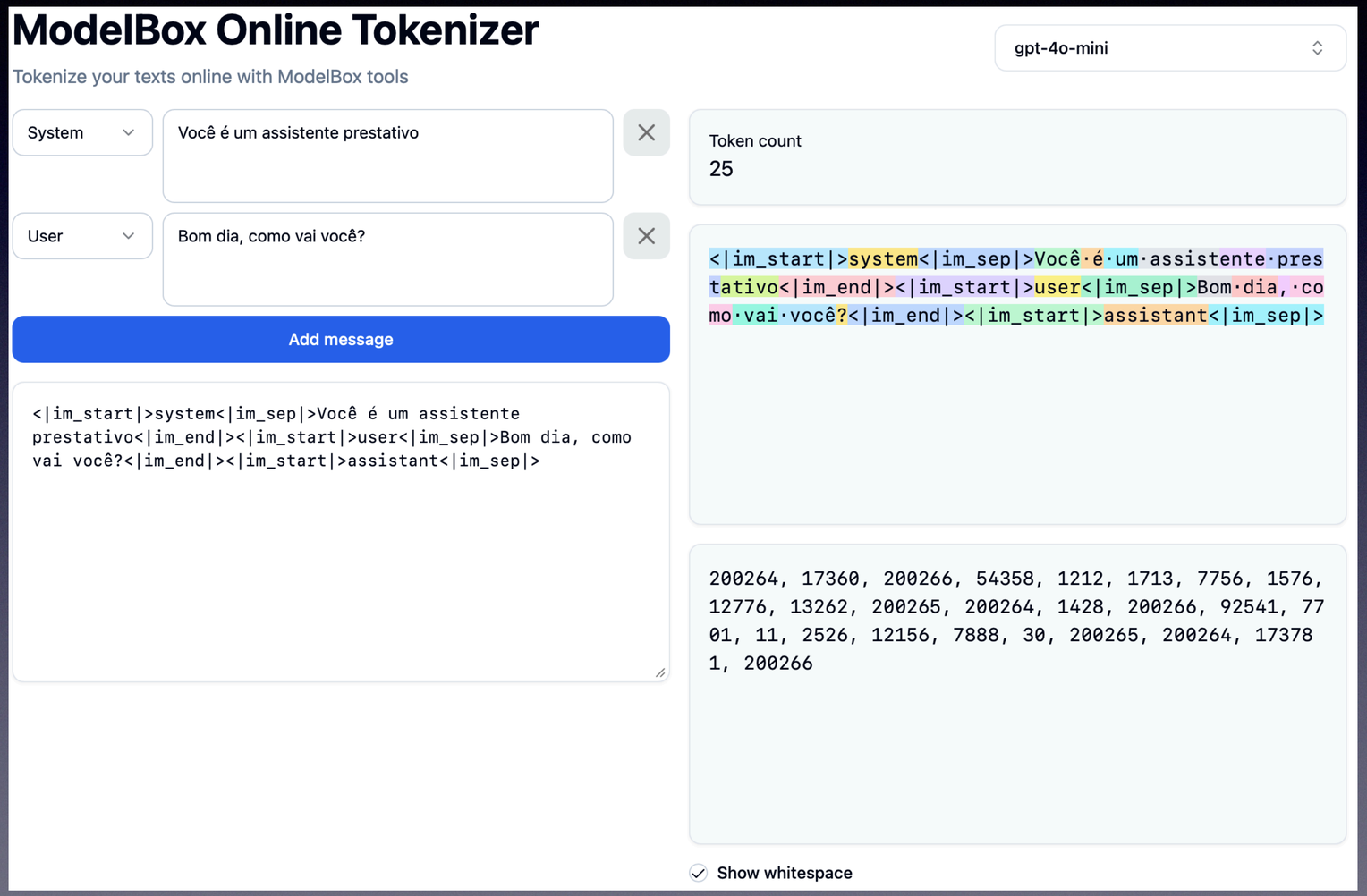
Quantidade máxima total de entrada + saída

aegro

Tokens

O que são tokens?

- ✦ Unidades básicas de processamento de LLMS
- ✦ Diferentes de palavras ou caracteres
- ✦ Varia por tokenizador de modelo



10 palavras → 25 tokens

Visualizador de tokens: https://tokenizer.model.box/

aegro

Tokens

Por que isso importa

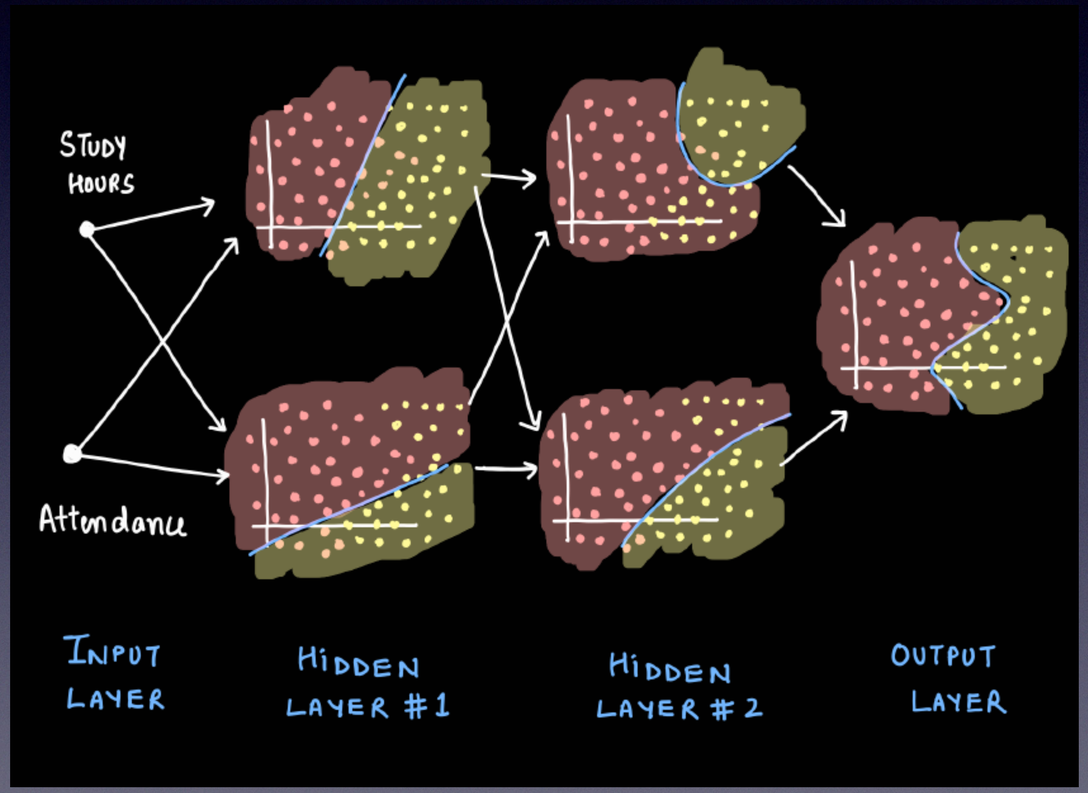
- ✦ Afeta os limites do contexto
- ✦ Influencia os cálculos de custo
- ✦ Impacta o desempenho do modelo

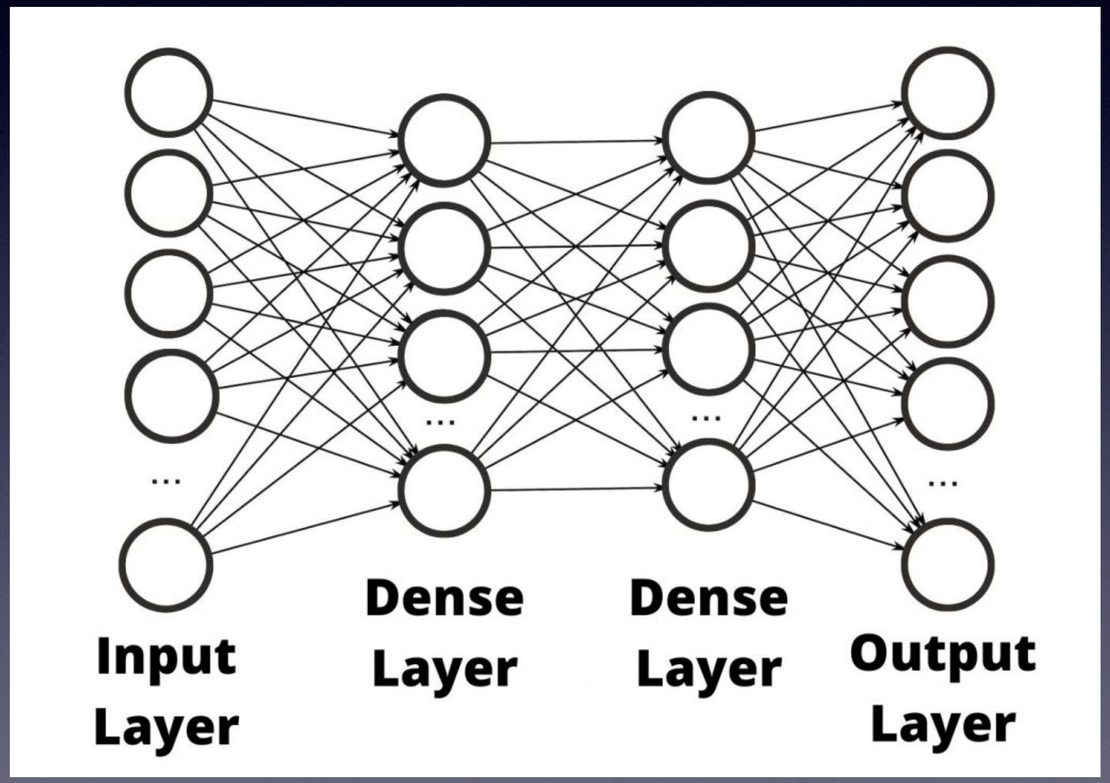
Dica prática

- ✦ Seja conciso com as informações importantes
- ✦ Use uma linguagem eficiente em termos de tokens

aegro

Camadas de Atenção





aegro

Janela de Contexto

Definição de Janela de Contexto

- Quantidade máxima de texto que o modelo pode considerar de uma só vez
- Varia de acordo com o modelo (de 2 mil a mais de 200 mil tokens)
- Inclui tanto a entrada quanto a saída gerada

aegro

Janela de Contexto

Estratégias de Gerenciamento de Contexto

- Priorizar informações importantes
- Usar resumos para documentos extensos
- Dividir tarefas complexas em partes menores
- Incluir código e documentação relevantes

aegro

Janela de Contexto

Impacto no Mundo Real

- Determina a quantidade de código que você pode analisar de uma só vez
- Afeta a continuidade da conversa
- Influencia a complexidade da tarefa

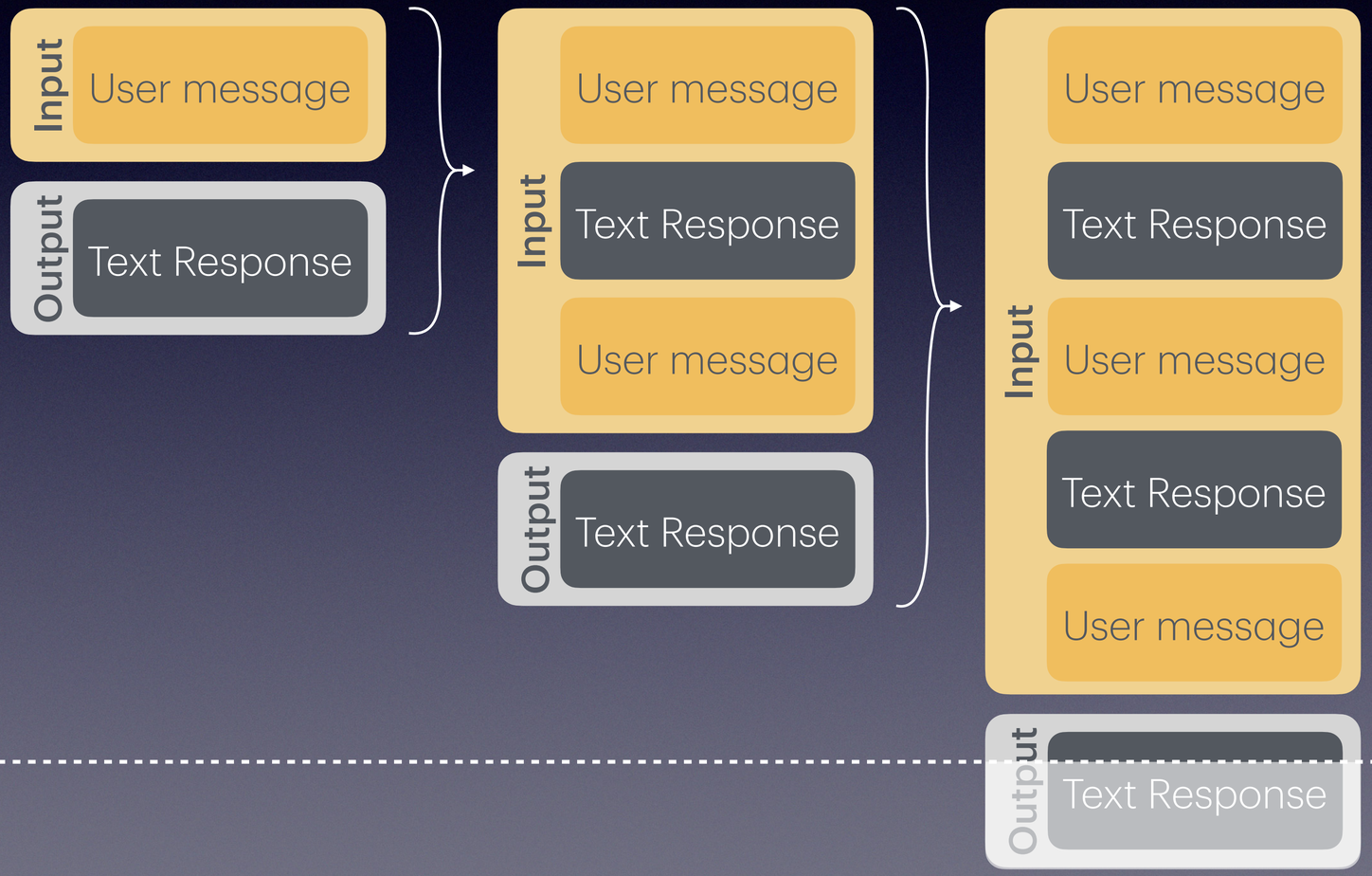
aegro

Janela de Contexto

aegro

Janela de Contexto

0 tokens

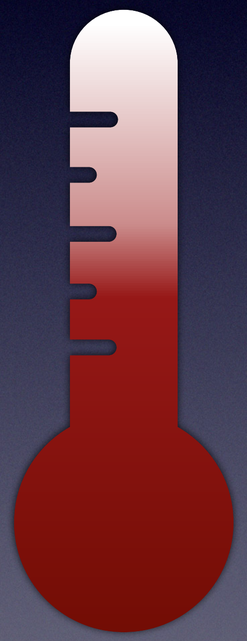


Janela de Contexto

200k tokens

aegro

Temperatura



Parâmetro de Temperatura

- ✦ Controla a aleatoriedade nas respostas do modelo
- ✦ **Intervalo:** 0,0 a 2,0 (tipicamente 0,0 a 1,0)

Efeitos da Temperatura

- ✦ **Baixa (0,0-0,3):** Saídas determinísticas e consistentes
- ✦ **Média (0,4-0,7):** Equilíbrio entre criatividade e confiabilidade
- ✦ **Alta (0,8-1,0+):** Saídas criativas e diversas

Recomendações de Desenvolvimento

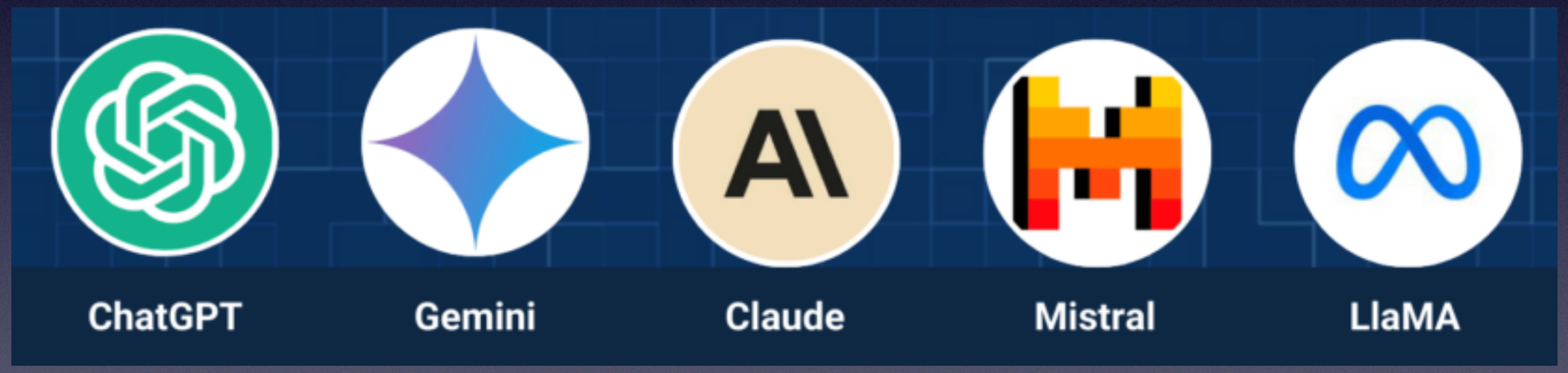
Geração de código: Temperatura baixa (0,1-0,3)

Brainstorming: Temperatura média (0,5-0,7)

Soluções criativas: Temperatura alta (0,7-0,9)

aegro

LLMs Utilizadas em Desenvolvimento



aegro

Demonstração

Modelos diferentes, resultados diferentes…

“Crie um endpoint REST API para autenticação de usuários”

aegro

Fundamentos da Engenharia de Prompt

O que é “Engenharia de Prompt”?

A arte e a ciência de criar instruções eficazes para obter os resultados desejados de LLMs.

aegro

Fundamentos da Engenharia de Prompt

Por que é importante?

- ✦ Impacta diretamente a qualidade da saída
- ✦ Reduz a iteração e o refinamento
- ✦ Permite a conclusão de tarefas complexas
- ✦ Maximiza as capacidades do modelo

aegro

Fundamentos da Engenharia de Prompt

Prompts bons vs. Ruins

Ruim: "Acelere isso"

Bom: "Otimize esta consulta ao banco de dados para obter o melhor desempenho.
Concentre-se em reduzir o tempo de execução da consulta e minimizar a carga do banco de dados."

aegro

Princípios de Prompts Eficazes

Seja Específico 1

- Instruções claras e detalhadas
- Evite ambiguidades
- Especifique o formato desejado

Forneça Contexto 2

- Inf. relevantes sobre o contexto
- Situação/ambiente atual
- Tentativas ou soluções anteriores

Definir o Formato 3

- Estrutura de saída esperada
- Linguagem e estilo de código
- Requisitos documentados

Utilizar Exemplos 4

- Exemplos de entrada/saída
- Problemas semelhantes
- Padrões de código

aegro

Engenharia de Prompt

aegro

Prompt
Zero-Shot

- Nenhum exemplo fornecido
- Depende de conhecimento pré-treinado

`“Crie uma função em Python para validar endereços de e-mail"`

aegro

Prompt
One-Shot

- Um exemplo fornecido
- Ajuda o modelo a entender as expectativas

```
“Crie funções de validação de email como neste exemplo:

def validate_phone(phone):
    return bool(re.match(r'^\d{10}$', phone))

Agora crie validate_email(email)”
```

aegro

Prompt
Few-Shot

- Múltiplos exemplos
- Estabelece padrões claros

```
“Seguindo esses padrões, crie validate_date():
  [Exemplos de várias funções de validação]”
```

aegro

Prompt
Role-Based

- Aplicar uma funcionalidade (Persona)

“Aja como um engenheiro de segurança sênior.
Analise este código em busca de
vulnerabilidades e sugira melhorias.”

aegro

Prompt
Step-by-Step Instructions

- Lista de Comandos/Tarefas Passo a Passo

```
“1. Analise o fluxo de autenticação atual
  2. Identifique as falhas de segurança
    3. Proponha melhorias específicas
  4. Forneça exemplos de implementação”
```

aegro

Prompt Template-Based

- Aplicar um Modelo em um Prompt

“Gere uma função [linguagem] que [ação]
com [requisitos].
Inclua tratamento de erros e documentação.”

aegro

Prompt Chain-of-Thought

O que é Cadeia de Raciocínio?

- Solicitar ao modelo que pense passo a passo
- Decompor problemas complexos
- Mostrar o processo de raciocínio

```
"Problema: Corrigir esta consulta lenta ao banco de dados

Vamos pensar passo a passo:
1. Primeiro, analisar a estrutura atual da consulta
2. Identificar possíveis gargalos
3. Verificar se há índices ausentes
4. Considerar técnicas de otimização de consultas
5. Implementar e testar melhorias"
```

Benefícios

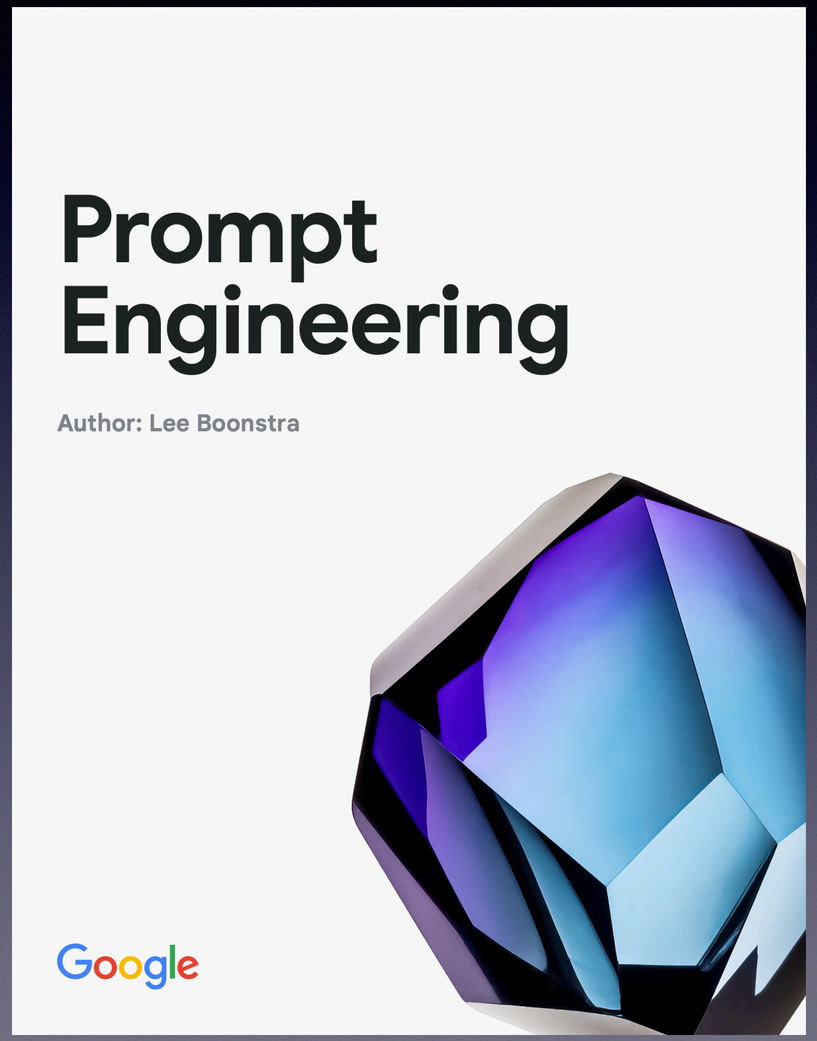
- Melhor resolução de problemas
- Menos erros
- Raciocínio mais transparente
- Depuração mais fácil

aegro

Prompt - Documentação

https://github.com/kushsengar/Prompt-Engineering-By-Google

https://aistudio.google.com



aegro

Engenharia de Contexto

aegro

Engenharia de Contexto

O que é Engenharia de Contexto?

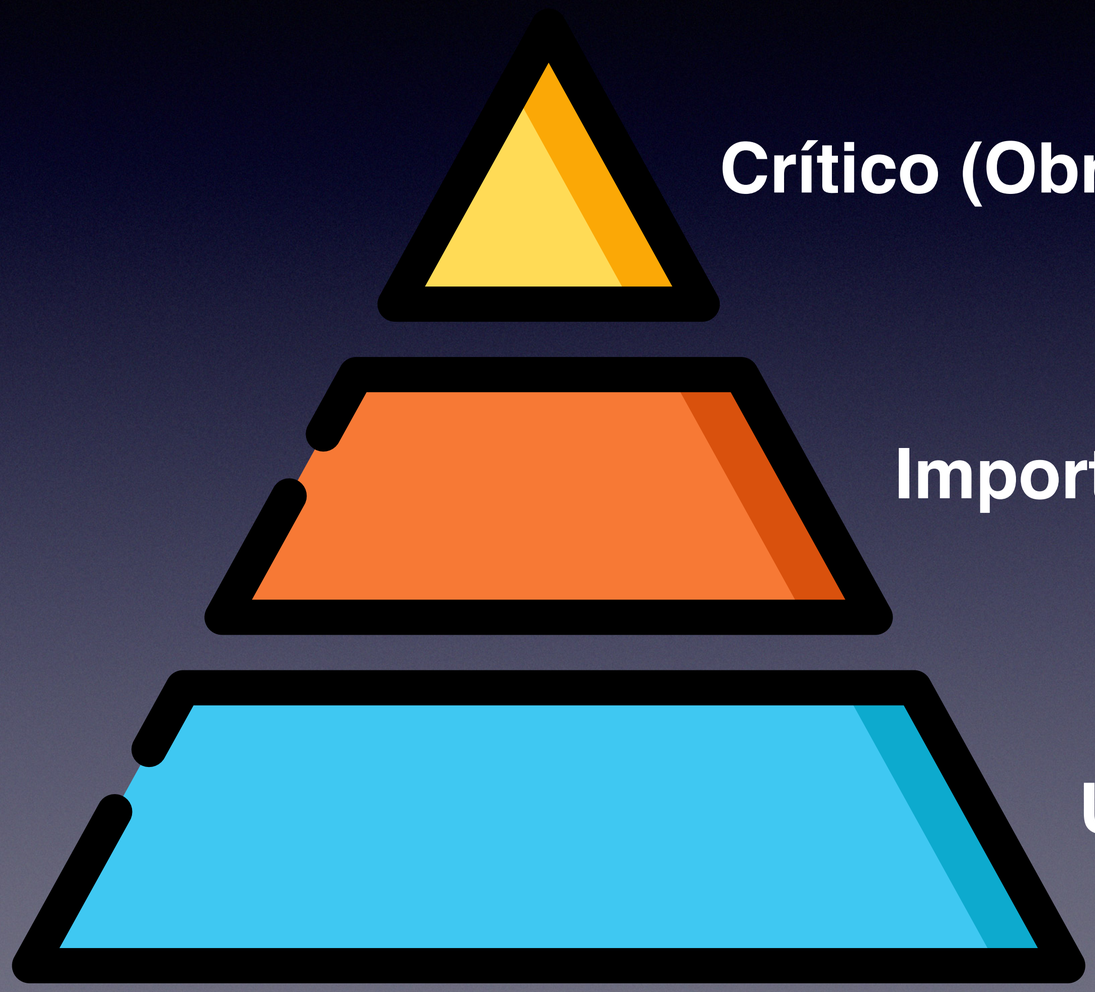
Gestão estratégica das informações fornecidas ao LLM para otimizar as respostas e trabalhar dentro das limitações.

Desafios

- Janelas de contexto limitadas
- Sobrecarga de informações
- Determinação de relevância
- Otimização de custos

aegro

Pirâmide de Gestão de Contexto



Crítico (Obrigatório)

Importante (Recomendado)

Útil (Desejado)

aegro

Estratégias de Gestão de Contexto

Estrutura de Priorização

Crítico: Requisitos principais, restrições

Importante: Padrões de projeto, dependências

Útil: Exemplos, código relacionado

aegro

Estratégias de Gestão de Contexto

Resumo de Informações

Condensar documentos extensos

Extrair pontos-chave

Manter detalhes essenciais

Fragmentação Estratégica

Dividir grandes bases de código em módulos

Focar em seções relevantes

Incluir importações/ dependências relacionadas

Apresentação/ Documentação Progressiva

Começar com o contexto geral

Adicionar detalhes conforme necessário

Manter a continuidade da conversa

aegro

Estratégias para Contexto de Arquivos

Análise de arquivos únicos:

Arquivo: `autenticacao.py`
- Incluir todo o conteúdo do arquivo
- Adicionar uma breve descrição
- Observar as dependências e importações

Projetos com múltiplos arquivos:

Estrutura de projeto:

```
src/
|——aut/
| |——login.py (arq principal)
| |——models.py (relacionado)
| |——utils.py (dependências)
```

Boas Práticas:

- ✦ Comece com o arquivo de ponto de entrada
- ✦ Inclua as dependências relacionadas
- ✦ Adicione uma visão geral da arquitetura
- ✦ Observe os padrões de projeto utilizados

aegro

Gestão de Limite de Contexto

O que acontece quando você atinge os limites de contexto?

Sintomas

Respostas são cortadas
O modelo perde o fio da conversa
A qualidade se degrada

Soluções

Resumir o contexto anterior
Focar apenas na tarefa atual
Usar resumos de arquivos para bases de código grandes
Dividir em prompts menores e mais focados

aegro

Geração de Código na Prática

aegro

Fluxo de Desenvolvimento com IA

Requisitos para Codificação

Crie um sistema de cadastro de usuários com:

- Validação de e-mail
- Criptografia de senha
- Criação de perfil
- Envio de e-mail de boas-vindas

Padrões de Geração de Código

Implementação de funções a partir de especificações

- Criação de endpoints de API
- Projeto de esquema de banco de dados
- Geração de casos de teste

Refinamento Iterativo

Revisão do código gerado

- Identificação de melhorias
- Solicitação de modificações específicas
- Validação da funcionalidade

aegro

Cenários de Geração de Código

Implementação de Função

Crie uma função JavaScript que:

- Valide números de cartão de crédito usando o algoritmo de Luhn
- Retorne um valor booleano de validade
- Inclua tratamento de erros completo
- Possua documentação JSDoc

Integração de API

Implemente uma rota Node.js Express que:

- Conecte-se à API de pagamento do Stripe
- Processe pagamentos
- Inclua tratamento de erros
- Retorne os códigos de status apropriados

Operações de Banco de Dados

Gere consultas SQL para um sistema de gerenciamento de usuários:

- Crie uma tabela de usuários com as restrições adequadas
- Insira usuários com verificação de e-mails duplicados
- Atualize o perfil do usuário com segurança
- Exclua usuários com opções em cascata

Técnicas Avançadas
Desenvolvimento Iterativo

Não espere perfeição logo no início

Geração Inicial

Requisitos Claros
Implementação Básica

→

Revisão e Identificação

Análise do Código Gerado
Problemas e Melhorias

→

Aprimoramento Direcionado

Solicitar Alterações Específicas
Foco em Áreas Problemáticas

→

Validação

Testar a Implantação
Verificar a Funcionalidade

Exemplo de Iteração

```
Você: "Crie uma função de ordenação"
IA: [Ordenação básica bubble]
Você: "Torne-a mais eficiente para grandes conjuntos de dados"
IA: [Implementa o quicksort]
Você: "Adicione suporte para funções de comparação personalizadas"
IA: [Aprimora com suporte a callbacks]
```

aegro

Fim do 1º Dia

Dúvidas?

aegro