

INSTITUTO INFNET

ESCOLA SUPERIOR DE TECNOLOGIA

GRADUAÇÃO EM CIÊNCIA DE DADOS



Engenharia de Prompts para Ciência de Dados
[24E4_4]

TP 2

Alunos: Gustavo Carneiro Alves.

2024

Link github: [gustavo_alves_DR4_TP2](#)

Link collab:  DR4_TP2.ipynb

Exercício 1

Princípios de Criação de Prompts Eficazes

Alguns princípios são essenciais para a construção de prompts claros e obtenção de respostas precisas. Identifique e explique quais princípios foram utilizados no prompt abaixo:

Prompt

Como um especialista em turismo, liste 3 atrações turísticas imperdíveis de Paris e forneça uma breve descrição para cada uma delas. Aqui está um exemplo do formato esperado:

###

Atração 1: Torre Eiffel

Descrição: Um dos monumentos mais icônicos do mundo, conhecido por sua estrutura de ferro e vista panorâmica da cidade.



R.

- Descrição da Tarefa (Instruções) “liste 3 atrações...”
- Informação de Fundo (Persona, Contexto) “Como um especialista em turismo...”
- Exemplos “Aqui está um exemplo...”

Exercício 2

Simulação de Atendimento ao Cliente com Diferentes Personas

Vamos simular uma situação de telemarketing onde persona e tom são críticos para o sucesso da interação. Para criarmos uma base de treinamento dos operadores, simule duas respostas para o mesmo atendimento, ora com um cliente agressivo, ora com um cliente tranquilo. Crie e teste um prompt (**Poe.com**) para cada situação a partir do atendimento:

- o “Bom dia, me chamo João e lhe trago hoje uma promoção imperdível sobre a sua assinatura de internet móvel. Por mais 20 reais, consigo lhe oferecer 10GB a mais na sua franquia. Quando podemos estar agendando a sua migração?”

Descreva o comportamento do cliente nas duas situações? As respostas saíram como o esperado?

"Bom dia, me chamo João e lhe trago hoje uma promoção imperdível sobre a sua assinatura de internet móvel. Por mais 20 reais, consigo lhe oferecer 10GB a mais na sua franquia. Quando podemos estar agendando a sua migração?" - simule duas respostas para o mesmo atendimento, ora com um cliente agressivo, ora com um cliente tranquilo.

GPT-4o Poe

Claro! Aqui estão duas simulações de resposta:

Cliente Agressivo

"Olha, João, já estou cansado dessas ligações! Não estou interessado em pagar mais nada. Já pago um absurdo por esse serviço que mal funciona. Não me ligue mais!"

Cliente Tranquilo

"Oi, João! Obrigado pela oferta. Vou pensar a respeito e verificar se essa quantidade extra de dados realmente me atende. Posso entrar em contato com vocês depois para decidir?"

R. Ambas as respostas estão de acordo com o esperado. Só acabaram sendo um pouco curtas demais.

Exercício 3 Automação de Mensagens Promocionais com Princípios de Prompting

A empresa de turismo viagens.com deseja automatizar o processo de criação de envio de mensagens para clientes, informando-lhes sobre promoções recentes. Escreva um prompt que contenha os três princípios de 1) persona, 2) dados e 3) tom para gerar a mensagem automática. Execute o prompt usando a sua conta Poe.com e justifique se o LLM respondeu como esperado.

"Crie uma mensagem automática como se fosse de uma empresa de turismo, falando sobre promoções recentes, que seja amigável e prestativa, mantenha um tom acolhedor e profissional #nome da empresa: InfnetTour #promoção: desconto de 20% em todas as viagens"

"Crie uma mensagem automática como se fosse de uma empresa de turismo, falando sobre promoções recentes, que seja amigável e prestativa, mantenha um tom acolhedor e profissional #nome da empresa: InfnetTour #promoção: desconto de 20% em todas as viagens"

GPT-4o Poe

Assunto: 🌍 Descubra o Mundo com 20% de Desconto na InfnetTour! ⭐

Olá!

Aqui é da **InfnetTour**, e temos uma novidade incrível para você! Estamos oferecendo um desconto especial de **20% em todas as nossas viagens**. É a oportunidade perfeita para realizar aquela viagem dos sonhos, explorar novos destinos e criar memórias inesquecíveis.

Seja uma escapada de fim de semana ou uma aventura internacional, temos opções para todos os perfis. Nossa equipe está pronta para ajudar você a planejar cada detalhe com conforto e segurança.

Entre em contato conosco para saber mais sobre nossos pacotes e serviços. Estamos ansiosos para tornar sua próxima viagem uma experiência maravilhosa!

Atenciosamente,

Equipe InfnetTour



Se precisar de mais informações, estamos sempre à disposição!

R. Respondeu como o esperado, foi gentil, usou os dados corretamente e se passou pela empresa.

Exercício 4 Desenvolvimento de Prompts para Recomendação de Produtos

Você está desenvolvendo um sistema de recomendações de produtos para um e-commerce. Crie dois prompts com base nos **princípios de prompting**, sendo o primeiro um prompt simples e o segundo um **prompt com exemplos** (few-shot prompting). Utilize o **Poe.com** para testar os prompts com um LLM, comparando os resultados gerados. Qual dos prompts foi mais eficaz e por quê?

GPT-4o Poe

Claro! Aqui estão algumas sugestões de produtos populares para um e-commerce:

Tecnologia

- Fones de Ouvido Bluetooth
- Smartwatches
- Carregadores Portáteis
- Tablets

Casa e Cozinha

- Air Fryers
- Máquinas de Café Expresso
- Purificadores de Ar
- Kits de Organização

Moda

- Tênis Confortáveis
- Relógios Elegantes
- Bolsas de Couro
- Óculos de Sol

Beleza e Cuidados Pessoais

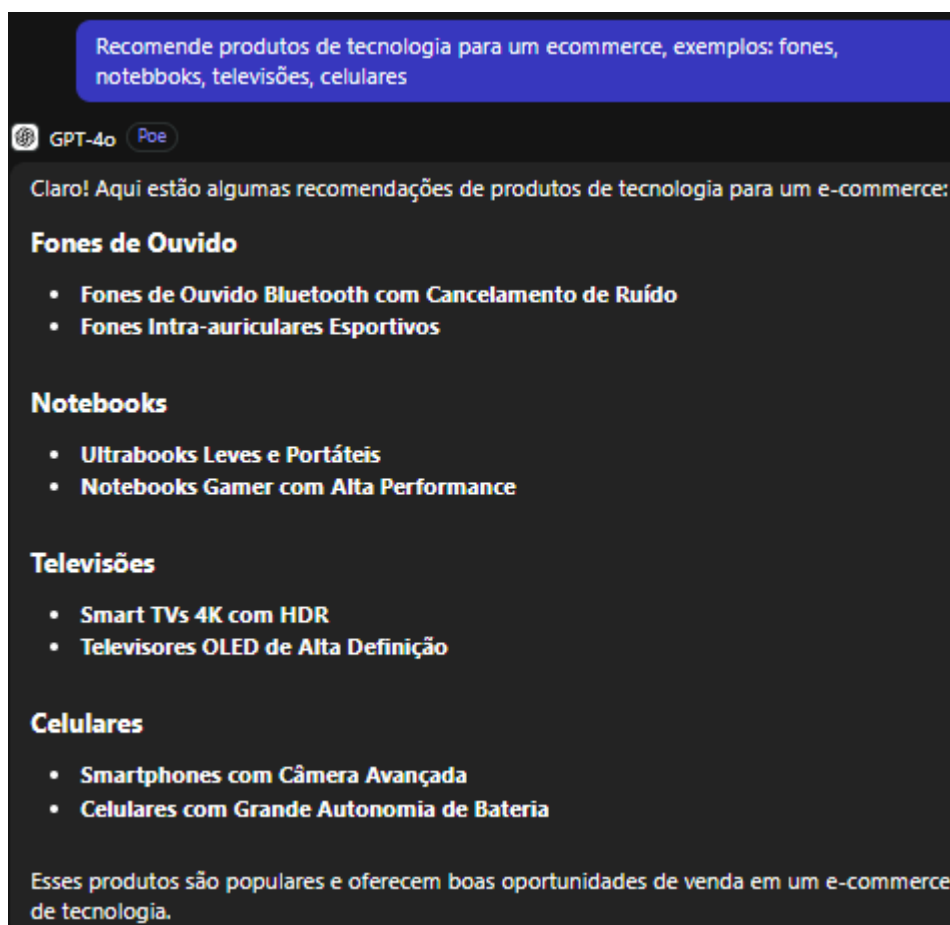
- Skincare Kits
- Aparelhos de Barba e Cabelo
- Perfumes Importados
- Maquiagem Vegana

Esportes e Lazer

- Roupas de Academia
- Bicicletas Dobráveis
- Equipamentos de Camping
- Suplementos Nutricionais

Livros e Entretenimento

- Romances Best-sellers
- Jogos de Tabuleiro
- E-books
- Álbuns de Música



R. O segundo pq ele respondeu diretamente sobre o nicho especificado.

Exercício 5	Listagem de Componentes de Computadores de Alto Desempenho
<p>Utilize o Poe.com para criar um prompt simples que faça o LLM listar os principais componentes de um computador de alto desempenho, suas capacidades computacionais (HD, RAM, CPU, GPU...), marcas, modelos e preços. Teste o prompt e explique se o resultado atendeu às expectativas: as marcas e modelos existem? As capacidades do computador são de alto desempenho? Compare com fontes encontradas na internet.</p>	

liste os principais componentes de um computador de alto desempenho da atualidade (2024), suas capacidades computacionais (HD, RAM, CPU, GPU...), marcas, modelos e preços.

GPT-4o Poe ...

Claro! Aqui estão alguns dos principais componentes de um computador de alto desempenho em 2024, com exemplos de marcas, modelos e capacidades:

Processador (CPU)

- **Intel Core i9-14900K**
 - **Cores/Threads:** 16/32
 - **Frequência:** Até 5.8 GHz
 - **Preço:** Aproximadamente \$600
- **AMD Ryzen 9 7950X**
 - **Cores/Threads:** 16/32
 - **Frequência:** Até 5.7 GHz
 - **Preço:** Aproximadamente \$550

Placa de Vídeo (GPU)

- **NVIDIA GeForce RTX 5090**
 - **VRAM:** 24 GB GDDR6X
 - **Preço:** Aproximadamente \$2000
- **AMD Radeon RX 7900 XT**
 - **VRAM:** 20 GB GDDR6
 - **Preço:** Aproximadamente \$1000

Memória RAM

- **Corsair Vengeance DDR5**
 - **Capacidade:** 64 GB (2x32 GB)
 - **Frequência:** 6000 MHz
 - **Preço:** Aproximadamente \$300
- **G.Skill Trident Z5 RGB DDR5**
 - **Capacidade:** 64 GB (2x32 GB)
 - **Frequência:** 6400 MHz
 - **Preço:** Aproximadamente \$350

- **G.Skill Trident Z5 RGB DDR5**
 - Capacidade: 64 GB (2x32 GB)
 - Frequência: 6400 MHz
 - Preço: Aproximadamente \$350

Armazenamento (SSD)

- **Samsung 990 Pro NVMe SSD**
 - Capacidade: 2 TB
 - Velocidade de Leitura/Escrita: Até 7450/6900 MB/s
 - Preço: Aproximadamente \$250
- **WD Black SN850X NVMe SSD**
 - Capacidade: 2 TB
 - Velocidade de Leitura/Escrita: Até 7300/6600 MB/s
 - Preço: Aproximadamente \$240

Placa-mãe

- **ASUS ROG Strix Z790-E Gaming**
 - Preço: Aproximadamente \$400
- **MSI MEG X670E Godlike**
 - Preço: Aproximadamente \$700

Fonte de Alimentação

- **Corsair RM1000x (2024)**
 - Capacidade: 1000W, 80 Plus Gold
 - Preço: Aproximadamente \$200

Cooler

- **Noctua NH-D15 Chromax.Black**
 - Preço: Aproximadamente \$100
- **Corsair iCUE H150i Elite Capellix XT Liquid CPU Cooler**
 - Preço: Aproximadamente \$180

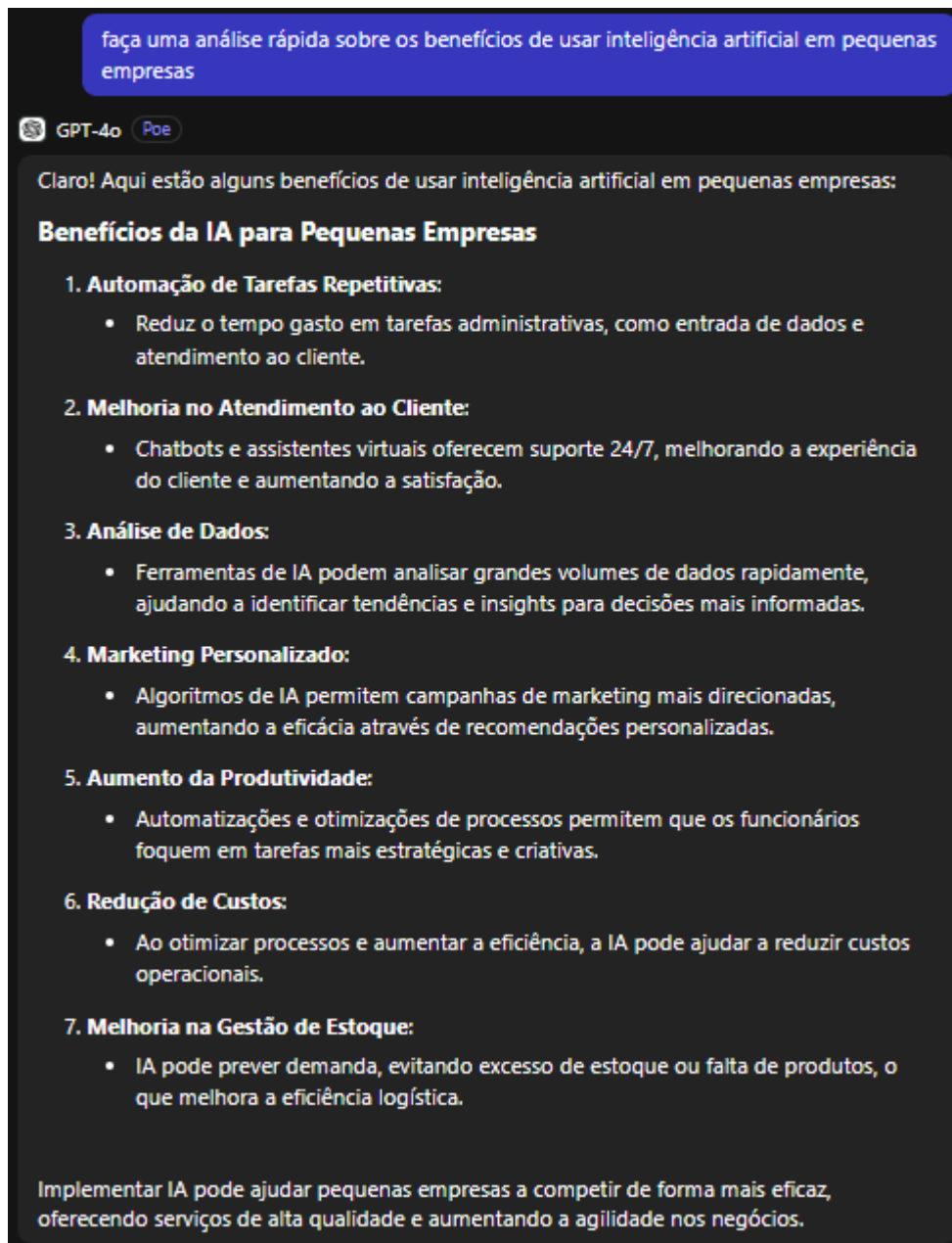
Esses componentes são exemplos de produtos de alto desempenho disponíveis em 2024, adequados para jogos, edição de vídeo e outras tarefas que exigem alta capacidade computacional. Os preços podem variar conforme o mercado.

R. Foi uma ótima resposta com componentes realmente de topo de linha e preços condizentes com a realidade em dólar.

Exercício 6

Análise dos Benefícios da Inteligência Artificial para Pequenas Empresas

Desenvolva um **prompt simples** para que o LLM forneça uma análise rápida sobre os benefícios de usar inteligência artificial em pequenas empresas. Qual foi o resultado gerado pelo LLM? Aplique os princípios de prompt para obter resultados mais precisos. Teste os prompts na sua conta do **Poe.com** e copie os prompts e respectivas saídas, junto com a explicação do que foi feito entre um prompt e outro.



faça uma análise rápida sobre os benefícios de usar inteligência artificial em pequenas empresas

GPT-4o (Poe)

Claro! Aqui estão alguns benefícios de usar inteligência artificial em pequenas empresas:

Benefícios da IA para Pequenas Empresas

- 1. Automação de Tarefas Repetitivas:**
 - Reduz o tempo gasto em tarefas administrativas, como entrada de dados e atendimento ao cliente.
- 2. Melhoria no Atendimento ao Cliente:**
 - Chatbots e assistentes virtuais oferecem suporte 24/7, melhorando a experiência do cliente e aumentando a satisfação.
- 3. Análise de Dados:**
 - Ferramentas de IA podem analisar grandes volumes de dados rapidamente, ajudando a identificar tendências e insights para decisões mais informadas.
- 4. Marketing Personalizado:**
 - Algoritmos de IA permitem campanhas de marketing mais direcionadas, aumentando a eficácia através de recomendações personalizadas.
- 5. Aumento da Produtividade:**
 - Automatizações e otimizações de processos permitem que os funcionários foquem em tarefas mais estratégicas e criativas.
- 6. Redução de Custos:**
 - Ao otimizar processos e aumentar a eficiência, a IA pode ajudar a reduzir custos operacionais.
- 7. Melhoria na Gestão de Estoque:**
 - IA pode prever demanda, evitando excesso de estoque ou falta de produtos, o que melhora a eficiência logística.

Implementar IA pode ajudar pequenas empresas a competir de forma mais eficaz, oferecendo serviços de alta qualidade e aumentando a agilidade nos negócios.

faça uma análise rápida sobre os benefícios de usar inteligência artificial em pequenas empresas como se fosse um especialista em IA querendo vender a integração com o chat GPT4. Evite tópicos e tente ser formal e profissional.

GPT-4o Poe

Certamente! Aqui está uma análise formal sobre os benefícios de integrar inteligência artificial, especificamente o Chat GPT-4, em pequenas empresas.

A integração de inteligência artificial, como o Chat GPT-4, nas operações de pequenas empresas oferece uma transformação significativa em termos de eficiência e competitividade. Como especialista em IA, é claro que a automação e a personalização são duas das principais vantagens que essa tecnologia proporciona.

Em primeiro lugar, o Chat GPT-4 permite a automação de atendimento ao cliente, garantindo suporte contínuo e respostas rápidas a consultas comuns. Essa capacidade não apenas melhora a experiência do cliente, mas também libera recursos humanos para se concentrarem em questões mais complexas que exigem um toque pessoal.

Além disso, o uso de IA para análise de dados pode proporcionar insights valiosos que impulsionam a tomada de decisões estratégicas. A capacidade do Chat GPT-4 de interpretar grandes volumes de dados textuais e fornecer relatórios detalhados permite que as empresas identifiquem padrões de consumo e oportunidades de mercado com rapidez e precisão.

O marketing personalizado é outro benefício crucial. Com o Chat GPT-4, as empresas podem criar campanhas de marketing mais precisas, baseadas em interações e preferências do cliente, aumentando o retorno sobre o investimento e melhorando a fidelização.

Por fim, a eficiência operacional é amplamente aprimorada. O Chat GPT-4 pode ser integrado a sistemas internos para otimizar processos, reduzir erros e diminuir custos operacionais. Essa eficiência é vital para pequenas empresas que buscam maximizar seus recursos limitados.

Em resumo, a implementação do Chat GPT-4 oferece uma solução robusta para desafios comuns enfrentados por pequenas empresas, promovendo crescimento sustentável e inovação contínua.

No segundo prompt optei por estabelecer uma persona e um tom para a conversa. Isso fez com que a resposta parecesse mais humana e menos robótica como se viesse de uma matéria online.

Exercício 7 Resumo de Notícia Utilizando Exemplos em Prompts

Usando a API Gemini, crie um **notebook** que utilize prompts para resumir uma notícia (escolha da página principal de um portal de notícias e copie no notebook). O prompt deve solicitar um resumo dessa notícia usando o princípio de Exemplos para guiar a resposta do LLM. Teste o código e avalie a qualidade do resumo gerado.

```
[ ] # Instalação da biblioteca para interagir com a API Gemini
!pip install google-generativeai
```

```
# Importação da biblioteca
import google.generativeai as genai
from google.colab import userdata
API_KEY = userdata.get('GOOGLE_API_KEY')

# Autenticação (substitua pela sua chave de API)
genai.configure(api_key=API_KEY)
```

```
# Notícia (substitua pelo texto da notícia que você deseja resumir)
noticia = """
Celso Araujo Sampaio de Novais, de 41 anos, foi atingido por um tiro de fuzil nas costas e estava internado na Unidade de Terapia Intensiva (UTI) do Hospital Geral de Guarulhos.

A comunicação do óbito foi registrada no 7ª DP da cidade, de acordo com a Secretaria de Segurança Pública de São Paulo.

Além de Sampaio, outras duas pessoas foram atingidas: um homem, de 39 anos, que também foi internado no Hospital Geral de Guarulhos, e uma mulher, de 28 anos, que recebeu atendimento médico e foi liberada em seguida - ela foi
Foram ao menos 27 disparos efetuados por dois criminosos, segundo informações da perícia do Departamento de Homicídios e Proteção à Pessoa Humana (DHPP) da Polícia Civil, que investiga o caso.
"""

# Prompt com exemplos para guiar o resumo
prompt = f"""
**Instruções:** Resuma a notícia abaixo em 1 sentença usando suas próprias palavras.

**Exemplo 1:**
Notícia: "O governo anunciou hoje um novo plano de saúde para os cidadãos."
Resumo: "O governo lançou um novo plano de saúde."

**Exemplo 2:**
Notícia: "A empresa X anunciou que irá fechar 500 vagas de trabalho."
Resumo: "A empresa X irá demitir 500 funcionários."

**Notícia:**
{noticia}

**Resumo:**
"""

# Executando o prompt com o modelo Gemini
model = genai.GenerativeModel("gemini-1.5-flash")
response = model.generate_content(prompt)

# Exibindo a resposta gerada
print(f"Resumo gerado pelo LLM: {response}")
print(response)
```

```
Resumo gerado pelo LLM:
response:
GenerateContentResponse(
  done=True,
  iterator=None,
  result=protos.GenerateContentResponse({
    "candidates": [
      {
        "content": {
          "parts": [
            {
              "text": "Um homem de 41 anos morreu ap\u00f3s ser atingido por um tiro de fuzil durante um ataque de dois criminosos que dispararam 27 tiros em Guarulhos. \n"
            }
          ],
          "role": "model"
        },
        "finish_reason": "STOP",
        "index": 0,
        "safety_ratings": [
          {
            "category": "HARM_CATEGORY_SEXUALLY_EXPLICIT",
            "probability": "NEGLIGIBLE"
          },
          {
            "category": "HARM_CATEGORY_HATE_SPEECH",
            "probability": "NEGLIGIBLE"
          },
          {
            "category": "HARM_CATEGORY_HARASSMENT",
            "probability": "LOW"
          },
          {
            "category": "HARM_CATEGORY_DANGEROUS_CONTENT",
            "probability": "LOW"
          }
        ]
      }
    ],
    "usage_metadata": {
      "prompt_token_count": 304,
      "candidates_token_count": 37,
      "total_token_count": 341
    }
  })
)
```

R.

Exercício 8 Identificação de Entidades em Notícias com LLM

Uma consultoria lhe contratou para automatizar a descoberta de pessoas mencionadas em notícias de jornal. Com base no Exercício 7, escolha 3 notícias e monte uma aplicação com um prompt para o LLM identificar menções a diferentes entidades em cada notícia (como pessoas, órgãos públicos, empresas...). Implemente um **notebook** para testar o código usando a API do Gemini. Descreva o prompt, seus resultados e avalie se a resposta do modelo atende às expectativas da consultoria.

```
[5] # Notícias escolhidas
noticia = """
Notícia 1: O presidente da República, Luiz Inácio Lula da Silva, anunciou ontem que o governo vai investir R$ 1 bilhão em infraestrutura para o estado de São Paulo.
Notícia 2: A empresa de tecnologia, Google, anunciou que vai criar 1.000 novos empregos no Brasil nos próximos dois anos.
Notícia 3: O ministro da Educação, Abraham Bragança, disse que o governo vai aumentar o investimento em educação em 10% no próximo ano.
...

# Prompt para o LLM
prompt = """Identifique as menções a diferentes entidades (pessoas, órgãos públicos, empresas, etc.) na seguinte notícia:"

**Notícia:**
{noticia}

...

# Executando o prompt com o modelo Gemini
model = gemini.GenerativeModel("gemini-1.5-flash")
response = model.generate_content(prompt)

# Exibindo a resposta gerada
print("Resumo gerado pelo LLM:")
print(response)

Resumo gerado pelo LLM:
response:
GenerateContentResponse(
  done=True,
  iterator=None,
  result=proto.GenerateContentResponse({
    "candidates": [
      {
        "content": {
          "parts": [
            {
              "text": "## Entidades mencionadas nas notícias:\n\n**Notícia 1:**\n\n**Pessoas:**\n\n* Luiz Inácio Lula da Silva (Presidente da República)\n\n**Órgãos Públicos:**\n\n* Governo\n\n**Entidades Geográficas:**\n\n* Estado de São Paulo\n\n**Notícia 2:**\n\n**Empresas:**\n\n* Google\n\n**Países:**\n\n* Brasil\n\n**Notícia 3:**\n\n**Pessoas:**\n\n* Abraham Bragança (Ministro da Educação)\n\n**Órgãos Públicos:**\n\n* Governo\n\n"
            }
          ]
        },
        "role": "model",
        "finish_reason": "STOP",
        "index": 0,
        "safety_ratings": [
          {
            "category": "HARM_CATEGORY_SEXUALLY_EXPLICIT",
            "probability": "NEGLIGIBLE"
          },
          {
            "category": "HARM_CATEGORY_HATE_SPEECH",
            "probability": "NEGLIGIBLE"
          },
          {
            "category": "HARM_CATEGORY_HARASSMENT",
            "probability": "NEGLIGIBLE"
          }
        ]
      }
    ]
  })
)
```

Resposta: """## Entidades mencionadas nas notícias:\n\n**Notícia 1:**\n\n**Pessoas:**\n\n* Luiz Inácio Lula da Silva (Presidente da República)\n\n**Órgãos Públicos:**\n\n* Governo\n\n**Entidades Geográficas:**\n\n* Estado de São Paulo\n\n**Notícia 2:**\n\n**Empresas:**\n\n* Google\n\n**Países:**\n\n* Brasil\n\n**Notícia 3:**\n\n**Pessoas:**\n\n* Abraham Bragança (Ministro da Educação)\n\n**Órgãos Públicos:**\n\n* Governo\n\n"

Exercício 9

Cálculo de Tokens em Texto Longo com API Gemini

Implemente um **notebook** que use a API Gemini para calcular a quantidade de **tokens** necessários para processar um texto de 5.000 palavras. Baseie-se no modelo de tokenização utilizado por Gemini e explique como a quantidade de tokens influencia o custo e o desempenho da interação com LLMs em textos longos.

```
▶ model = genai.GenerativeModel("gemini-1.5-flash")

# Prompt para o LLM
prompt = f"""
Água mole em pedra dura, tanto bate até que fura.
"""*500

print(f"Quantidade de palavras: {len(prompt.split())}")

# Executando o prompt com o modelo Gemini
response = model.generate_content(prompt)

# Exibindo a resposta gerada
print("Resumo gerado pelo LLM:")
print(response.usage_metadata)
```

↗

```
Quantidade de palavras: 5000
Resumo gerado pelo LLM:
prompt_token_count: 7002
candidates_token_count: 158
total_token_count: 7160
```

Exercício 10

Otimização de Respostas com Role Prompting

No **Poe.com**, aplique a técnica de **Role Prompting** para otimizar as respostas do LLM Claude3.5. O cenário é o seguinte: você está desenvolvendo um assistente virtual para uma empresa de consultoria jurídica. Crie um prompt onde o modelo deve **assumir o papel de um advogado especializado em direito contábil** ao responder perguntas sobre Imposto de Renda de Pessoa Física. Avalie a resposta do modelo para uma mesma pergunta sobre IRPF num prompt com e sem Role Prompting.

Claude-3.5-Haiku Poe

A restituição do IRPF (Imposto de Renda Pessoa Física) é um processo no qual o contribuinte pode receber de volta o valor do imposto que pagou a mais durante o ano-base, após fazer a declaração do Imposto de Renda.

Principais características:

1. Quando acontece:

- Após a entrega da declaração de Imposto de Renda
- Quando o valor retido durante o ano é maior que o imposto devido

2. Quem pode receber:

- Pessoas físicas que declararam o Imposto de Renda
- Contribuintes que têm direito a restituição após análise da declaração

3. Como funciona:

- O contribuinte entrega a declaração
- A Receita Federal analisa as informações
- Se confirmado o valor a restituir, é liberado o pagamento

4. Formas de recebimento:

- Depósito em conta bancária
- Ordem bancária para saque em agências
- Transferência para outras contas

5. Ordem de pagamento:

- Prioridade para idosos, deficientes físicos e professores

A restituição representa um possível "troco" do imposto pago durante o ano fiscal.

me explique O que é a restituição do IRPF como se fosse um advogado especializado em direito contábil

Claude-3.5-Haiku Poe

Como advogado especializado em direito contábil, posso esclarecer que a restituição do IRPF configura-se como um mecanismo jurídico-tributário de recomposição patrimonial, previsto na legislação fiscal brasileira, especificamente regulamentado pela Instrução Normativa RFB nº 2112/2022.

Aspectos jurídicos fundamentais:

🔍 **Natureza Jurídica:**

- Direito subjetivo do contribuinte
- Modalidade de restituição de tributos pagos indevidamente
- Instrumento de concretização do princípio constitucional da capacidade contributiva

📄 **Elementos Constitutivos:**

- Declaração de Ajuste Anual
- Comprovação de recolhimento a maior
- Análise fiscal pela Receita Federal

📋 **Requisitos Legais:**

- Entrega tempestiva da declaração
- Regularidade documental
- Ausência de pendências fiscais
- Conformidade com a legislação tributária vigente

📋 **Processamento Técnico:**

- Cruzamento de dados declarados
- Verificação de possíveis inconsistências
- Validação dos valores declarados
- Apuração do montante restituível

Trata-se, portanto, de um instituto jurídico-tributário de natureza restitutória, fundamentado no princípio da não-cumulatividade fiscal.

Deseja que eu detalhe algum aspecto específico dessa explicação técnica?

R. No prompt com role prompting a resposta foi mais técnica e mais difícil de ser compreendida, no entanto passou mais confiabilidade.

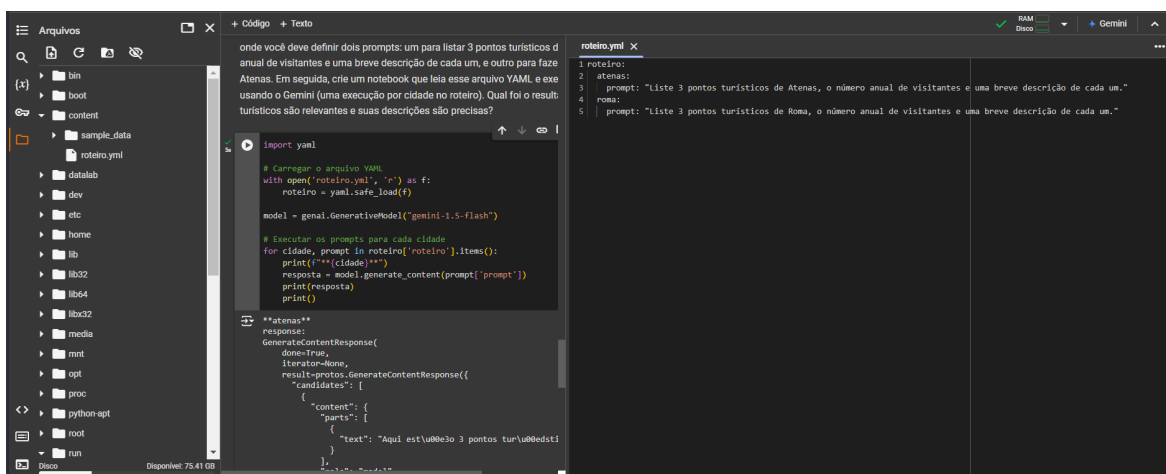
Exercício 11

Estruturação de Prompts com Arquivo YAML

Crie um **arquivo YAML** que represente a estrutura:

```
{'roteiro': {'atenas': 'prompt', 'roma': 'prompt'}}
```

onde você deve definir dois prompts: um para listar 3 pontos turísticos de Roma, o número anual de visitantes e uma breve descrição de cada um, e outro para fazer o mesmo com Atenas. Em seguida, crie um **notebook** que leia esse arquivo YAML e execute os prompts usando o Gemini (uma execução por cidade no roteiro). Qual foi o resultado? Os pontos turísticos são relevantes e suas descrições são precisas?



```
Arquivos
{X}
  bin
  boot
  content
  sample_data
  roteiro.yaml
  databab
  dev
  etc
  home
  lib
  lib32
  lib64
  libx32
  media
  mnt
  opt
  proc
  python-apt
  root
  run
  Disco

+ Código + Texto
roteiro.yaml
roteiro:
1
2
3
4
5
atenas:
prompt: "Liste 3 pontos turísticos de Atenas, o número anual de visitantes e uma breve descrição de cada um."
roma:
prompt: "Liste 3 pontos turísticos de Roma, o número anual de visitantes e uma breve descrição de cada um."

Código
In [ ]:
# Carregar o arquivo YAML
with open('roteiro.yaml', 'r') as f:
    roteiro = yaml.safe_load(f)

model = genai.GenerativeModel("gemini-1.5-flash")

# Executar os prompts para cada cidade
for cidade, prompt in roteiro['roteiro'].items():
    print(f"***{cidade}***")
    resposta = model.generate_content(prompt['prompt'])
    print(resposta)
    print()

**atenas**
resposta:
GenerateContentResponse(
  done=True,
  iterator=None,
  result=protos.GenerateContentResponse({
    "candidates": [
      {
        "content": {
          "parts": [
            {
              "text": "Aqui est\u00e3o 3 pontos tur\u00edsticos de Atenas:
1. Acr\u00f3poli: O principal s\u00edmbolo da civiliza\u00e7\u00e3o grega antiga, abrigando o Parten\u00f5n e outros templos importantes.
2. Col\u00edneo: Um dos maiores e mais antigos templos gregos, dedicado a Apolo.
3. Teatro de Dion\u00edsio: Um dos maiores teatros antigos do mundo, usado para performances de drama grego.
O n\u00famero anual de visitantes \u00e9 de aproximadamente 2,5 milh\u00f5es de pessoas.
Uma breve descri\u00e7\u00e3o de cada um:
- Acr\u00f3poli: Um complexo de edif\u00edcios religiosos e pol\u00edticos no topo de uma colina.
- Col\u00edneo: Um templo com colunas j\u00f4nicas.
- Teatro de Dion\u00edsio: Um teatro semi-circular com uma orquestra no topo da inclina\u00e7\u00e3o."
            }
          ]
        }
      }
    ]
  })

***roma***
resposta:
GenerateContentResponse(
  done=True,
  iterator=None,
  result=protos.GenerateContentResponse({
    "candidates": [
      {
        "content": {
          "parts": [
            {
              "text": "Aqui est\u00e3o 3 pontos tur\u00edsticos de Roma:
1. Col\u00edseu: Um dos maiores e mais antigos templos gregos, dedicado a Apolo.
2. Col\u00edneo: Um dos maiores e mais antigos templos gregos, dedicado a Apolo.
3. Teatro de Dion\u00edsio: Um dos maiores teatros antigos do mundo, usado para performances de drama grego.
O n\u00famero anual de visitantes \u00e9 de aproximadamente 2,5 milh\u00f5es de pessoas.
Uma breve descri\u00e7\u00e3o de cada um:
- Acr\u00f3poli: Um complexo de edif\u00edcios religiosos e pol\u00edticos no topo de uma colina.
- Col\u00edneo: Um templo com colunas j\u00f4nicas.
- Teatro de Dion\u00edsio: Um teatro semi-circular com uma orquestra no topo da inclina\u00e7\u00e3o."
            }
          ]
        }
      }
    ]
  })
```

Exercício 12

Sistema de Envio de Emails com Dados Tur\u00edsticos em JSON

Queremos montar um sistema que envie emails com os pontos tur\u00edsticos de cidades num roteiro de viagem. Com base no **Exerc\u00edcio 11**, altere o prompt para estruturar a sa\u00edda em formato JSON, com descri\u00e7\u00f5es e n\u00famero de visitantes. Crie um **notebook** que use a biblioteca pandas para converter esse JSON para dataframe e plot um \u00fanico gr\u00e1fico de barras horizontais com todos os pontos tur\u00edsticos, destacando as cores das barras de acordo com a cidade do ponto tur\u00edstico (Coliseu, em Roma deve ter uma cor diferente que Parthenon, em Atenas). Coloque as legendas no gr\u00e1fico.

roteiro:

atenas:

prompt: "Liste 3 pontos turísticos de Atenas, o número anual de visitantes e uma breve descrição de cada um. Faça com que a resposta seja um json. Os visitantes anuais são um número sem ponto ou vírgula. Evite o fórum romano. Exemplo: nome:acropole, visitantes_anuais:320000, desc:antigo sitio"

roma:

prompt: "Liste 3 pontos turísticos de Roma, o número anual de visitantes e uma breve descrição de cada um. Faça com que a resposta seja um json. Os visitantes anuais são um número sem ponto ou vírgula. Evite o fórum romano. Exemplo: nome:acropole, visitantes_anuais:320000, desc:antigo sitio"

```
...
roteiro:
atenas:
prompt: "Liste 3 pontos turísticos de Atenas, o número anual de visitantes e uma breve descrição de cada um. Faça com que a resposta seja um json. Os visitantes anuais são um número sem ponto ou vírgula. Evite o fórum romano. Exemplo: nome:acropole, visitantes_anuais:320000, desc:antigo sitio"
roma:
prompt: "Liste 3 pontos turísticos de Roma, o número anual de visitantes e uma breve descrição de cada um. Faça com que a resposta seja um json. Os visitantes anuais são um número sem ponto ou vírgula. Evite o fórum romano. Exemplo: nome:acropole, visitantes_anuais:320000, desc:antigo sitio"

...

with open('roteiro.json', 'r') as f:
    roteiro = json.load(f)

# Instancia o GPT do OpenAI
model = openai.OpenAI(api_key=os.getenv('OPENAI_API_KEY'))

response_list = []
for cidade, prompt in roteiro.items():
    response = model.generate_completion(prompt)
    entrada = response.last
    entrada = entrada.replace('```', '')
    entrada = entrada.replace('```', '')
    entrada = entrada.replace('```', '')
    response_list.append(entrada)
    print(entrada)

# Exemplo de saída
[[{"nome": "Acropolis", "visitantes_anuais": 320000, "desc": "Antigo sítio com os restos de vários templos e monumentos, incluindo o Partenão."}, {"nome": "Museu da Acropolis", "visitantes_anuais": 100000, "desc": "Museu que abriga artefatos e"}, {"nome": "Coliseu", "visitantes_anuais": 100000, "desc": "Um antigo anfiteatro romano, um dos monumentos mais famosos e maiores de Roma, conhecido por suas lutas de gladiadores e espetáculos grandiosos."}, {"nome": "Museu do Vaticano", "visitantes_anuais": 100000, "desc": "Museu que abriga artefatos e"}]]

# Exemplo de saída
[[{"nome": "Acropolis", "visitantes_anuais": 320000, "desc": "Antigo sítio com os restos de vários templos e monumentos, incluindo o Partenão."}, {"nome": "Museu da Acropolis", "visitantes_anuais": 100000, "desc": "Museu que abriga artefatos e"}, {"nome": "Coliseu", "visitantes_anuais": 100000, "desc": "Um antigo anfiteatro romano, um dos monumentos mais famosos e maiores de Roma, conhecido por suas lutas de gladiadores e espetáculos grandiosos."}, {"nome": "Museu do Vaticano", "visitantes_anuais": 100000, "desc": "Museu que abriga artefatos e"}]]

# Exemplo de saída
[[{"nome": "Acropolis", "visitantes_anuais": 320000, "desc": "Antigo sítio com os restos de vários templos e monumentos, incluindo o Partenão."}, {"nome": "Museu da Acropolis", "visitantes_anuais": 100000, "desc": "Museu que abriga artefatos e"}, {"nome": "Coliseu", "visitantes_anuais": 100000, "desc": "Um antigo anfiteatro romano, um dos monumentos mais famosos e maiores de Roma, conhecido por suas lutas de gladiadores e espetáculos grandiosos."}, {"nome": "Museu do Vaticano", "visitantes_anuais": 100000, "desc": "Museu que abriga artefatos e"}]]
```

