

Elaine Cristina C. Ferreira

Bootcamp Microsoft AI for Tech - OpenAI Services

20 de fevereiro de 2025

### Objetivo Geral:

O objetivo deste desafio é explorar o desenvolvimento de aplicações práticas utilizando o Azure OpenAI, incluindo chamadas de API e integração com o Semantic Kernel.

### O que fazer?

- Explore a demonstração apresentada no conteúdo prático ou faça um resumo do que aprendeu neste conteúdo. Em seguida armazene em um repositório no GitHub e compartilhe o link na entrega do desafio.

### .Conteúdo:

O curso proporcionou uma introdução abrangente a Cotas e Limites:

LLMs geralmente possuem **cotas de uso**, que podem ser **limites de tokens** (unidades de texto) processados ou **número de usuários** acessando o modelo. Essas cotas existem para garantir a **qualidade do serviço**, evitar o uso excessivo e permitir a geração adequada de tokens. Gerenciar as cotas envolve entender os recursos, os diferentes modelos, os custos e os impactos de aumentar os limites.

## 2. Filtros de Conteúdo:

Para garantir o uso responsável da IA, LLMs utilizam **filtros de conteúdo** para controlar a saída gerada. Esses filtros visam evitar a geração de conteúdo inadequado, como:

- **Ódio e imparcialidade:** Conteúdo discriminatório ou preconceituoso.
- **Sexual:** Conteúdo explícito ou sugestivo.
- **Violência:** Conteúdo que promova ou descreva violência.
- **Automutilação e ataques indiretos:** Conteúdo que incentive a autoagressão.
- **Jailbreak:** Tentativas de burlar as restrições do modelo.

É crucial revisar e ajustar os filtros regularmente para garantir sua eficácia.

## 3. Proteção e Segurança:

Além dos filtros, outras medidas de proteção e segurança incluem:

- **Inclusão de termos:** Bloquear palavras ou frases específicas.
- **Anotação e bloqueio de usuários:** Identificar e restringir usuários abusivos.
- **Níveis de severidade:** Definir a intensidade da filtragem.
- **Filtros de entrada e saída:** Controlar tanto o que entra quanto o que sai do modelo.

#### 4. Modelagem de Problemas:

LLMs podem ser usados para modelar e resolver problemas do mundo real. Isso significa transformar um problema em um formato que o LLM possa entender e processar.

#### 5. Prompt Engineering:

Prompt engineering é a arte de construir prompts (instruções) eficazes para obter os resultados desejados de um LLM. Um bom prompt é claro, específico, contextualizado e define o formato de saída desejado.

Elementos importantes do prompt engineering:

- **System message:** Define o contexto inicial, comportamento e instruções básicas do modelo.
- **Few-shots:** Fornece exemplos de como o modelo deve responder.
- **Procheck:** Processamento cruzado para melhorar a qualidade da resposta.
- **Cues:** Dicas e subdicas para guiar o modelo.
- **Task Breakdown:** Divide tarefas complexas em etapas menores.
- **Chain of thought:** Mostra ao modelo como resolver problemas passo-a-passo.
- **Tree of thought:** Permite que o modelo explore múltiplas linhas de raciocínio.

## 6. DALL-E e outros modelos:

- DALL-E: Modelo para geração de imagens que requer prompts claros, específicos e contextuais.
- Chat vs. Não chat: O prompt engineering se aplica a ambos os tipos de modelos, mas com algumas diferenças.
- O1: Modelo de geração de relatórios que exige contexto detalhado, instruções e planejamento.

## 7. Exemplos de Prompts:

Construir prompts eficazes, incluindo o uso de RAG (Retrieval-Augmented Generation) para contextualização e análise de sentimentos.

Devemos ter atenção ao bom uso dos prompts:

Exemplo de descrição ruim para prompt:

"Me fale sobre gatos" - não tem descrição

Descrição boa: "Descreva 3 características únicas dos gatos domésticos e como elas beneficiam sua sobrevivência"

## Conclusão:

O curso proporcionou um panorama abrangente sobre o uso de LLMs, desde o gerenciamento de cotas até a construção de prompts eficazes, com foco na segurança, na responsabilidade e na otimização dos resultados.