Pontifícia Universidade Católica do Rio de Janeiro
Departamento de Informática

Projeto final de programação

Chatbot para Data Engineering

Júlia Araújo

Orientador: Marcos Kalinowski

Rio de Janeiro

Dezembro de 2024

Sumário

1 Breve descrição

Esse trabalho aborda o desenvolvimento de um chatbot especializado para a área de Data Engineering, baseado em tecnologias de Recuperação Aumentada (RAG) e Processamento de Linguagem Natural (PLN). A principal função do programa é atuar como um assistente virtual, fornecendo respostas contextuais e precisas a partir de uma base de documentos previamente carregada e validada. O chatbot foi concebido para transformar documentos técnicos estáticos em um recurso dinâmico, acessível e interativo, permitindo aos usuários obterem informações relevantes de forma rápida e eficiente.

1.1 Problema

Profissionais de Data Engineering frequentemente enfrentam desafios relacionados ao acesso eficiente a informações em grandes volumes de documentação técnica, como manuais, guias de boas práticas e tutoriais. A busca manual por informações específicas pode ser demorada, especialmente quando a documentação não está organizada ou quando é necessário lidar com materiais extensos. Esse cenário impacta negativamente a produtividade, a tomada de decisões e a capacidade de resolver problemas de forma eficaz.

1.2 Justificativa

Com o avanço das tecnologias de inteligência artificial e PLN, soluções interativas como chatbots têm se tornado uma alternativa viável para superar os desafios mencionados. Um chatbot especializado na área de Data Engineering permite que os profissionais tenham acesso rápido e direcionado a informações técnicas, reduzindo o tempo de busca e promovendo maior eficiência no trabalho. Além disso, o uso de uma base de dados previamente carregada garante a precisão e a confiabilidade das respostas geradas.

1.3 Objetivo

O objetivo principal do programa é desenvolver um chatbot interativo capaz de responder perguntas relacionadas à engenharia de dados de forma contextual, utilizando uma base de dados técnica previamente carregada. Busca-se fornecer uma ferramenta prática e eficiente que facilite a consulta e interpretação de informações técnicas, auxiliando profissionais e estudantes na resolução de problemas e no aprendizado contínuo.

1.4 Hipótese

Se um chatbot especializado em Data Engineering for implementado, utilizando uma base de dados previamente carregada e tecnologias de recuperação aumentada, será possível melhorar a eficiência dos usuários, reduzindo o tempo gasto na busca por informações e promovendo decisões mais rápidas e fundamentadas.

1.5 Funções Específicas

O programa oferece as seguintes funcionalidades específicas:

- Consulta a documentação técnica por meio de perguntas naturais.
- Localização de informações relevantes em grandes volumes de documentos técnicos.]
- Geração de respostas contextuais com base na base de dados previamente carregada.
- Suporte para arquivo com formato PDF
- Interface gráfica amigável para carregamento de documentos e interação com o chatbot.

1.6 Usuários-Alvo

O programa foi concebido para atender aos seguintes grupos de usuários:

- Profissionais de Data Engineering: Especialmente aqueles que trabalham na construção, manutenção e otimização, que necessitam de respostas rápidas sobre ferramentas, conceitos e melhores práticas.
- Estudantes de Engenharia de Dados: Universitários e iniciantes na área que precisam de suporte no aprendizado de temas complexos relacionados à engenharia de dados.
- Pesquisadores e Desenvolvedores de Sistemas de Dados: Que buscam informações técnicas detalhadas para implementar e testar soluções inovadoras na área.

1.7 Natureza do Programa

O programa é uma ferramenta funcional que demonstra a integração de tecnologias de recuperação aumentada e processamento de linguagem natural em um chatbot especializado. Embora esteja funcional, o projeto é considerado uma prova de conceito avançada, com potencial para expansão e aplicação em ambientes corporativos e educacionais.

1.8 Ressalvas

Algumas limitações e ressalvas do programa incluem:

- Dependência de uma base de dados previamente carregada, o que pode limitar a abrangência das respostas.
- Necessidade de hardware com capacidade de processamento suficiente para lidar com grandes volumes de dados.
- Suporte restrito a formatos de arquivos mais comuns, como PDF.
- As respostas são limitadas ao conteúdo presente na base de dados, podendo ser insuficientes para dúvidas fora do escopo.

2 Visão de Projeto

A visão de projeto deste chatbot busca delinear cenários que ilustrem seu funcionamento ideal, suas aplicações práticas e suas limitações conhecidas. Essa abordagem oferece diretrizes para o uso do programa, ajustes durante o desenvolvimento e melhorias futuras. Os cenários são divididos em positivos, demonstrando casos de sucesso, e negativos, que expõem limitações que podem ser trabalhadas para aprimorar a ferramenta.

2.1 Cenário Positivo 1: Resolução Rápida de Dúvidas

João, um profissional que frequentemente consulta materiais técnicos, utiliza o chatbot para esclarecer uma dúvida sobre um procedimento descrito em documentos previamente carregados. Ele insere a pergunta, o chatbot analisa a base de dados, composta por manuais técnicos, artigos e guias práticos, e retorna uma resposta estruturada. A interação permite que João resolva rapidamente sua dúvida, sem precisar buscar manualmente a informação nos materiais, otimizando seu tempo e garantindo maior precisão na execução da tarefa.

2.2 Cenário Positivo 2: Auxílio no Estudo de Conceitos Complexos

Mariana, uma estudante que está explorando novos conceitos técnicos, utiliza o chatbot para esclarecer um termo complexo encontrado em um dos documentos previamente carregados. O chatbot identifica a presença do termo em diferentes documentos e apresenta

uma explicação clara. Devido a organização e clareza da resposta, Mariana sente mais segurança para prosseguir com seus estudos, economizando o tempo que gastaria para encontrar e conectar essas informações manualmente.

2.3 Cenário Negativo 1: Limitações na Cobertura da Base de Dados

Carlos, um profissional, faz uma pergunta ao chatbot sobre um tópico específico, confiando que os documentos carregados contêm todas as informações necessárias. Ele insere uma pergunta, o chatbot analisa os documentos, mas não encontra nenhuma menção ao tópico solicitado. Em vez de fornecer uma resposta detalhada, o chatbot retorna uma mensagem informando que não há informações relevantes na base de dados carregada. Embora Carlos reconheça que a base de dados precisa ser enriquecida, ele fica limitado naquele momento e não consegue resolver sua dúvida. Esse cenário evidencia a importância de uma base de dados abrangente e continuamente atualizada.

2.4 Cenário Negativo 2: Ambiguidade nas Respostas

Ana, uma usuária com menos experiência, utiliza o chatbot para entender uma instrução contida nos documentos carregados. O chatbot retorna uma resposta baseada em múltiplos documentos, mas os trechos extraídos apresentam interpretações ligeiramente diferentes sobre o mesmo tema. A resposta, embora correta, é ambígua e deixa Ana confusa sobre qual abordagem seguir.

3 Documentação técnica do projeto

A seção de documentação técnica do chatbot especializado foi desenvolvida para orientar desenvolvedores, colaboradores e usuários técnicos interessados em entender ou reutilizar o programa. A seguir, são apresentados os detalhes organizados em seções específicas que abrangem os principais aspectos técnicos.

3.1 Especificação de Requisitos

3.1.1 Requisitos Funcionais

1. O chatbot deve permitir a consulta a uma base de dados previamente carregada.

2. O sistema deve responder a perguntas utilizando informações relevantes dos documentos.
3. A interface gráfica deve ser intuitiva, permitindo aos usuários interagir facilmente com o programa.
4. As respostas geradas devem incluir referências aos documentos utilizados.

3.1.2 Requisitos Não-Funcionais

1. A interface gráfica deve ser responsiva e funcionar em navegadores diferentes.
2. O sistema deve suportar múltiplos usuários simultaneamente, garantindo estabilidade em condições normais de uso.
3. O código deve ser modular e extensível, permitindo futuras integrações ou melhorias.
4. Deve haver suporte para idiomas, possibilitando a adaptação a diferentes contextos linguísticos.

3.2 Modelo de arquitetura

A arquitetura do chatbot foi projetada para ser modular, garantindo a separação clara das responsabilidades e a escalabilidade do sistema. Ela está dividida em quatro componentes principais, descritos abaixo:

3.2.1 Módulo de Carregamento e Indexação de Documentos

Este módulo é responsável por processar a base de dados fornecida pelo usuário. Ele organiza os documentos carregados em índices que facilitam a busca eficiente.

- Entrada: Documentos em formatos suportados, como PDF e texto simples.
- Processo: Indexação dos documentos utilizando técnicas de PLN (Processamento de Linguagem Natural).
- Saída: Dados indexados e prontos para consulta.

3.2.2 Módulo de Recuperação de Informações

Este módulo atua como o coração do sistema, processando consultas enviadas pelos usuários e localizando trechos relevantes nos documentos carregados.

- Entrada: Consultas em linguagem natural fornecidas pelos usuários.
- Processo: Busca nos índices da base de dados utilizando algoritmos de recuperação de informações, como FAISS.
- Saída: Trechos relevantes extraídos dos documentos.

3.2.3 Módulo de Geração de Respostas

Baseado nos trechos retornados pelo módulo de recuperação, este componente utiliza modelos de linguagem natural para sintetizar respostas claras e precisas.

- Entrada: Trechos relevantes dos documentos.
- Processo: Síntese de informações por meio de modelos de PLN.
- Saída: Respostas estruturadas e contextualizadas.

3.2.4 Módulo de Interface Gráfica

Este módulo é a camada de interação com o usuário, permitindo que perguntas sejam enviadas e respostas sejam visualizadas. Ele também fornece opções para carregar documentos e gerenciar a base de dados.

- Entrada: Interações do usuário, como perguntas ou carregamento de documentos.
- Processo: Envio de consultas para os módulos internos e exibição de respostas.
- Saída: Respostas e mensagens informativas exibidas para o usuário.

3.3 Modelo funcional

O funcionamento do programa pode ser dividido em três etapas principais:

1. Entrada de Dados
    - Usuário insere perguntas em linguagem natural por meio da interface gráfica.

- Documentos devem ser previamente carregados pelo administrador do sistema.

2. Processamento Interno
   - A consulta é analisada para identificar palavras-chave e contexto.
   - A base de dados indexada é pesquisada para encontrar trechos que correspondam à consulta.
   - Modelos de linguagem natural sintetizam as informações relevantes em uma resposta compreensível.
3. Saída de Dados
   - Respostas detalhadas e contextualizadas.

3.4 Código

- Linguagem: Python.
- Frameworks:
  - LangChain: Para integração de modelos de linguagem natural e pipelines de recuperação aumentada.
  - FAISS: Para indexação e busca eficiente.
  - Streamlit: Para desenvolvimento da interface gráfica interativa.

4 Manual de utilização

Este manual descreve as etapas para utilizar o programa, desde sua inicialização até a realização de consultas e interpretação de respostas. O programa foi projetado para facilitar o acesso a informações relevantes em documentos previamente carregados.

- Tarefa 1: Iniciar o Programa
  - Passo 1: Certifique-se de que todas as dependências do programa (como Python e bibliotecas necessárias) estão instaladas no sistema.
  - Passo 2: Abra o terminal no diretório onde o programa está localizado.
  - Passo 3: Execute o comando streamlit run projeto.py para iniciar a interface gráfica do programa.

  - Passo 4: Aguarde enquanto o Streamlit carrega o programa. O navegador será aberto automaticamente, exibindo a interface do programa.

- Tarefa 2: Fazer uma Pergunta ao Programa
  - Passo 1: Após iniciar o programa, verifique na interface se os documentos previamente carregados estão listados.
  - Passo 2: No campo de entrada de texto principal, digite sua pergunta.
  - Passo 3: Pressione "Enter" ou clique no botão de envio para processar sua consulta.
  - Passo 4: Aguarde enquanto o programa analisa os documentos e apresenta a resposta.
  - Passo 5: Leia a resposta exibida no painel principal, que incluirá trechos destacados do documento utilizado.

- Tarefa 3: Consultar Referências e Trechos Utilizados
  - Passo 1: Após obter a resposta à sua pergunta, role para baixo para visualizar os trechos utilizados.
  - Passo 2: Identifique as referências citadas, que indicam a página ou seção do documento de onde a informação foi extraída.
  - Passo 3: Clique no título do documento, se disponível, para abrir o texto original e confirmar as informações.

- Tarefa 4: Explorar Novas Consultas nos Mesmos Documentos
  - Passo 1: Insira novas perguntas no campo de texto, mesmo que tratem de temas diferentes dentro dos documentos carregados.
  - Passo 2: Aguarde enquanto o programa processa a nova consulta e apresenta a resposta correspondente.
  - Passo 3: Use as respostas como base para validar e expandir suas pesquisas, revisando as referências destacadas.

5 Conclusão

Este projeto apresenta uma abordagem prática e funcional para integrar técnicas de **Data Engineering** com modelos de linguagem natural em um chatbot voltado para a recuperação de informações textuais. O sistema implementa pipelines eficientes para

ingestão, fragmentação e indexação de documentos, permitindo consultas dinâmicas e contextualizadas.

Ao explorar conceitos fundamentais de engenharia de dados, como a transformação de dados não estruturados em conhecimento acessível, o projeto demonstra como soluções programáticas podem ser aplicadas para otimizar a interação humano-computador em grandes volumes de dados. Com potencial para expansão em ambientes de big data e aplicações documentais, o chatbot exemplifica como a programação pode ser utilizada para solucionar problemas práticos e melhorar o acesso a informações complexas.