Documentação do GPTZINHO

Tela de um aparelho eletrônico

Descrição gerada automaticamente com confiança média

Este código é uma implementação de uma aplicação de chat baseada no ChatGPT usando PyQt5 e a API da OpenAI. A aplicação permite ao usuário interagir com modelos de linguagem AI como GPT-3.5 e GPT-4 para simular uma interação com o chatGPT. Abaixo está uma descrição detalhada de cada parte do código.

Importações e Configurações Iniciais

Texto

Descrição gerada automaticamente

- Importações: São importadas várias bibliotecas necessárias para a interface gráfica (PyQt5), solicitações de rede (requests), manipulação de variáveis de ambiente (dotenv, os), e integração com a API da OpenAI.

- Configuração de ambiente: As variáveis de ambiente são carregadas de um arquivo `.env` que deve conter a chave API da OpenAI (`API\_KEY`).

Classes Principais

1. Worker (QThread)

Descrição:

Esta classe gerencia as solicitações assíncronas à API da OpenAI. Herda de `QThread` para executar operações em uma thread separada, evitando bloqueio da interface gráfica.

Texto

Descrição gerada automaticamente

Métodos

- `run()`: Executa a solicitação para a API da OpenAI e emite um sinal quando a resposta é recebida.

- `enviar\_solicitacao\_openai()`: Monta e envia a solicitação para a API da OpenAI, manipula a resposta e retorna os dados relevantes.

2. ChatApp (QWidget)

Texto

Descrição gerada automaticamente

Descrição:

Classe principal da aplicação que define a interface gráfica do usuário.

Atributos Importantes

- `modelo`: Armazena o modelo da API da OpenAI que está sendo usado.

- `temperatura`: Define a temperatura do modelo para geração de texto.

- `historico\_conversa`: Lista que armazena o histórico da conversa para manter o contexto nas solicitações à API.

Métodos

- `\_\_init\_\_()`: Construtor da classe que inicializa a interface gráfica.

- `set\_posicao\_visor()`, `set\_posicao\_botao\_aumentar\_temp()`, etc.: Métodos para ajustar o posicionamento e tamanho dos widgets.

- `aumentar\_temperatura()`, `diminuir\_temperatura()`: Ajustam a temperatura do modelo de IA.

- `mudar\_modelo\_gpt3()`, `mudar\_modelo\_gpt4()`: Alteram o modelo da API da OpenAI utilizado.

- `enviar\_mensagem()`: Envia a mensagem do usuário para a thread de processamento e atualiza a interface.

- `atualizar\_ui()`: Atualiza a interface com a resposta recebida da API.

- `limpar\_campos()`: Limpa os campos de entrada de texto e área de resultados.

Layout e Widgets

A interface é composta de várias áreas:

Área de entrada de texto:

Onde o usuário digita suas mensagens.

Área de exibição de resultados

Onde as mensagens do usuário e as respostas da AI são exibidas.

Contadores de tokens e caracteres

Mostra a contagem de tokens da mensagem e da resposta, além do total de caracteres digitados.

Botões de ação

Incluem botões para enviar a mensagem, limpar campos, ajustar temperatura e mudar entre modelos.

Método de Limpar Campo

Função `limpar\_campos()`

Este método é responsável por limpar a área de entrada de texto e a área de exibição de resultados.

Este método é conectado ao botão de limpar campos, permitindo que o usuário limpe rapidamente o texto digitado e as respostas exibidas.

Botão Limpar Campos

Se precisar de mais alguma coisa, estou à disposição!

Funcionamento

A aplicação inicia criando uma instância de `ChatApp` e a exibe. As interações do usuário são processadas por eventos conectados a métodos específicos que manipulam a lógica de envio de mensagens e atualização da interface.

Estilo e Aparência

A personalização da aparência dos widgets é realizada através de chamadas de método específicas que definem cores e estilos para diferentes componentes da interface.

Execução

Para rodar a aplicação, o código final precisa ser executado em um ambiente que tenha as dependências instaladas e uma chave API válida configurada no arquivo `.env`.

Conclusão

Esta aplicação é um exemplo robusto de como integrar tecnologias de AI em aplicações interativas baseadas em GUI, proporcionando uma plataforma para interações baseadas em texto com modelos de linguagem avançados.