O modelo escolhido foi o gemini-1.5-flash, importado diretamente do módulo google.generativeai. A escolha por importar o modelo via API no genai e não rodar localmente se deveu ao fato que nenhum dos modelos escolhidos do HuggingFace rodaram na minha máquina. Testei os seguintes modelos:

* TinyLlama/TinyLlama-1.1B-Chat-v1.0
* gpt2
* all-MiniLM-L6-v2
* meta-llama/Llama-3.1-8B
* meta-llama/Llama-3.2-1B
* distilbert-base-uncased

Todos eles deram erros variados de permissão, custo computacional inviável, erros na chamada da API, etc. Desta forma, optei pelo modelo Gemini que funcionou de forma razoável.

As vantagens na utilização deste modelo são:

**1. Desempenho**

Capacidade de Processamento: Modelos como o Gemini-1.5-Flash são otimizados para rapidez e eficiência em geração de texto. Isso significa que o chatbot poderá responder rapidamente a perguntas sobre reciclagem, melhorando a experiência do usuário.

Treinamento em Domínios Diversificados: Ele tem uma base de treinamento ampla e adaptável, o que pode garantir uma compreensão contextual adequada para o tema de reciclagem, mesmo sem ajuste fino.

Qualidade de Resposta: O modelo equilibra bem a qualidade das respostas com a velocidade de geração, sendo capaz de fornecer informações úteis e consistentes.

**2. Custo Computacional Local**

Eficiência: Comparado a modelos maiores, o Gemini-1.5-Flash consome menos recursos de memória e processamento, tornando-o mais viável para ser executado localmente em máquinas com recursos limitados.

Hardware acessível: Pode rodar em GPUs moderadas ou até mesmo em CPUs potentes, eliminando a necessidade de infraestrutura de alto custo.

Sustentabilidade: A redução no consumo energético também é alinhada aos princípios de sustentabilidade, especialmente relevante para o contexto da aplicação.

**3. Acessibilidade Local**

Independência de Nuvem: O modelo pode ser implementado localmente, permitindo total controle sobre os dados, o que é vantajoso para aplicações sensíveis que lidam com dados específicos de usuários ou regiões.

Privacidade: A operação local garante que os dados dos usuários não sejam transmitidos para servidores externos, aumentando a confiança na aplicação.

Disponibilidade: Sem dependência de serviços externos, o chatbot pode continuar operando mesmo em locais com conectividade limitada à internet, aumentando a acessibilidade da aplicação.