



# Documentação Técnica – IA Generativa Finetunada (GPT-2 - Signa)



## Arquivos do Projeto

Arquivo	Função
<code>gerador_de_modelo.py</code>	Script completo de pré-processamento e treino de um modelo GPT-2 com os dados institucionais da Signa
<code>signa.txt</code>	Arquivo gerado com o corpus textual (perguntas e respostas sobre a Signa) usado como base de treino
<code>API_gpt2_finetune.py</code>	API em FastAPI que recebe um prompt e retorna texto gerado pelo modelo finetunado
<code>requirements.txt</code>	Lista de bibliotecas necessárias para executar o treino e a API



## Lógica de Funcionamento



### Treinamento

1. O corpus `signa.txt` é carregado como lista de textos.
2. As entradas são tokenizadas com `distilgpt2` (modelo leve do GPT-2).
3. Cada entrada é usada como entrada e rótulo (self-supervised).
4. O modelo é treinado com `Trainer` por 3 épocas.
5. Ao final, o modelo é salvo com `save_pretrained()`.



### API

- Recebe um `prompt` via `POST /gerar`
- Usa o pipeline `text-generation` com o modelo treinado
- Retorna o texto gerado com base no prompt



## Como Executar


## 1. Treinar o modelo

```
python gerador_de_modelo.py
```

## 2. Iniciar a API

```
uvicorn API_gpt2_finetune:app --reload
```

### Observação Importante

 O teste web da API não pôde ser finalizado

Ao tentar publicar a API (Replit), o tamanho do arquivo `pytorch_model.bin` do modelo finetunado excedeu o limite de upload da plataforma.

Isso inviabilizou o deploy completo com geração online.

Recomenda-se rodar localmente ou subir em plataformas com suporte a arquivos > 100MB (ex: AWS S3, GCP, Azure, Hugging Face Pro).

### Exemplo de chamada via `curl`

```
curl -X POST http://localhost:8000/gerar \
-H "Content-Type: application/json" \
-d '{"prompt": "Pergunta: Quais serviços a Signa oferece?\nResposta:"}'
```

### Requisitos (requirements.txt)

```
transformers
datasets
torch
pydantic
```

---

## ✓ Pontos Positivos

- ✓ Total controle sobre os dados do modelo
  - ✓ Geração de texto mais natural, contínua e criativa
  - ✓ Funciona offline após o treino
  - ✓ API leve, sem dependência de OpenAI ou serviços externos
- 

## ⚠ Pontos a Melhorar

- ✗ Pode gerar respostas irrelevantes ou repetitivas se o prompt for curto
  - ⚠ Modelo pequeno (distilgpt2) tem limitações de entendimento
  - 🧠 Não sabe quando não sabe (pode "alucinar")
  - 🔁 Sem controle de contexto ou memória de diálogos
- 



## 🔬 Relatório de Testes

Prompt de entrada	Resultado gerado	Avaliação
"Pergunta: O que é a Signa? \nResposta:"	"A Signa é uma empresa portuguesa especializada..."	✓ Ótimo
"Pergunta: Como entrar em contato? \nResposta:"	"Você pode ligar para +351 214 127 780..."	✓ Correto
"Pergunta: Vocês atendem empresas fora de Portugal? \nResposta:"	"O site não informa atuação internacional..."	✓ Adequado
"Pergunta: Qual é o nome do fundador? \nResposta:"	"A Signa é especializada..."	⚠ Hallucinação
"A Signa oferece plano de saúde?"	"Sim, a Signa..."	✗ Invenção (não real)

---

## 💡 Melhorias Futuras

- 🔍 Incluir checagem de factualidade (RAG ou verificação por base vetorial)
- 📈 Usar GPT maior (gpt2-medium ou gpt2-xl)
- 🎯 Implementar filtros para detectar perguntas fora de escopo

-  Habilitar salvamento dos prompts e respostas geradas para avaliação contínua
-  Oferecer interface web para testes abertos (com streamlit ou Gradio)