

Configuração para desenvolvimento e execução do código

Este projeto tem como objetivo realizar a análise de dados relacionados à Uber utilizando técnicas de processamento de linguagem natural (PLN) e aprendizagem de máquina, rodando uma aplicação em Flask e utilizando a biblioteca Gradio do Python para uma visualização mais clara da interface. Para executar o projeto corretamente, siga as instruções detalhadas abaixo.

Vale citar que a versão final entregue como MVP está localizada na pasta Model_Sprint_5

1. Clonar o Repositório

Primeiro, é necessário clonar o repositório do projeto para o seu computador local. Abra um terminal e execute o seguinte comando:

```
git clone https://github.com/Intel-i-College/2024-1B-T10-SI06-G03.git
cd 2024-1B-T10-SI06-G03
```

2. Preparar o Ambiente e Baixar os Datasets

- Baixar os Datasets: Para rodar o notebook corretamente, você precisa baixar os datasets disponíveis no link a seguir: [Datasets](#). Os datasets necessários são:
 - cc.en.300.bin: [Download](#) - Este é um arquivo binário contendo palavras pré-treinadas para o FastText, uma biblioteca de vetorização de texto.
 - tweets_uber.csv: [Baixar Arquivo](#) - Este arquivo contém uma coleção de tweets relacionados à Uber. Note que este dataset foi modificado para fins de análise e não é o original fornecido pela Uber.
- Salvar os Datasets: Após baixar os arquivos, você deve colocá-los na pasta **data**, que está localizada em **src/Notebook/Model_Sprint5/data**. Para encontrar essa pasta, navegue pela estrutura de pastas do projeto até **src/Notebook/Model_Sprint5/data** e mova os arquivos baixados para lá.

3. Preparar o Ambiente de Desenvolvimento

- Instalar Dependências Necessárias para Rodar o Notebook: O projeto usa várias bibliotecas Python, listadas no arquivo **requirements.txt**. Para instalar todas elas de uma vez, abra um terminal, navegue até a pasta do projeto e execute o seguinte comando:

```
pip install -r requirements.txt
```

Este comando instalará todas as bibliotecas necessárias para rodar o notebook.

Alternativamente, você pode rodar a célula de "Instalação e Importação das Bibliotecas" no notebook para instalar as bibliotecas necessárias quando for rodar tudo.

- Instalar Dependências Necessárias para Rodar a Aplicação: Além das bibliotecas do notebook, a aplicação web requer algumas bibliotecas adicionais. Para instalá-las, execute o seguinte comando no terminal:

```
pip install gradio requests pandas matplotlib numpy flask flaskggr slack_bolt sqlalchemy
```

Este comando instalará as bibliotecas necessárias para rodar a interface Gradio e o servidor Flask.

4. Executar o Notebook

3. Abrir o Jupyter Notebook: Para abrir o notebook, navegue até a pasta `src/Notebook/Model_Sprint_5/` e abra o arquivo `Model.ipynb`.
4. Executar o Notebook: Dentro do Jupyter Notebook, execute todas as células clicando em `Executar tudo` ou `Run All` no menu. Este processo executará todo o código passo a passo. No final, este arquivo deve gerar dois arquivos na pasta `models`: `xgboost_negative_vs_rest.pkl` e `xgboost_positive_vs_neutral.pkl`. Além disso, deve gerar mais três arquivos na pasta `data`: `cc.en.300.vec`, `tweets_uber_vectorized.csv` e `tweets_uber.csv`. Esses arquivos são necessários para as etapas seguintes.

5. Executar a Interface Gradio

1. Abrir um Terminal: Abra um terminal e navegue até a pasta onde o script `gradio_interface.py` está localizado:

```
cd 2024-1B-T10-SI06-G03/src/Notebook/Model_Sprint_5/app
```

2. Executar o Script: Execute o script `gradio_interface.py` digitando o seguinte comando no terminal:

```
python gradio_interface.py
```

Isso iniciará a aplicação localmente em `http://127.0.0.1:7860/`. Abra um navegador e vá até esse endereço para ver a interface Gradio do projeto.

6. Executar o Servidor Flask

3. Abrir Outro Terminal: Abra outro terminal (ou uma nova aba no terminal atual) e navegue até a mesma pasta onde o script `run.py` está localizado:

```
cd 2024-1B-T10-SI06-G03/src/Notebook/Model_Sprint_5/app
```

4. Executar o Script: Execute o script `run.py` digitando o seguinte comando no terminal:

```
python run.py
```

Após executar os passos acima, ao clicar no link `http://127.0.0.1:7860/` no navegador, você deve ver a aplicação do projeto em funcionamento.