

Project 83: Chat with CSV (Q&A on Tabular Data)

Description:

This tool lets users upload a CSV file and ask natural language questions about its contents. It reads the file, summarizes structure, and uses LLM reasoning with few-shot examples to answer questions about totals, filters, trends, etc.

chat_with_csv.py

```
import os
import openai
import gradio as gr
import pandas as pd

# Load OpenAI API key
openai.api_key = os.getenv("OPENAI_API_KEY")

# Function to generate answer from CSV using OpenAI
def ask_csv_question(csv_file, query):
    try:
        # Read CSV into DataFrame
        df = pd.read_csv(csv_file.name)

        # Show summary for context
        sample = df.head(3).to_string()
        col_info = ", ".join(df.columns)

        # Build prompt
        prompt = (
            f"You are a data analyst assistant. Use the data below to answer t\n"
            f"Columns: {col_info}\n\n"
            f"Sample Data:\n{sample}\n\n"
            f"Question: {query}\nAnswer:"
        )

        # Call OpenAI
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo",
```

```

        messages=[{"role": "user", "content": prompt}]
    )

    return response["choices"][0]["message"]["content"].strip()

except Exception as e:
    return f"Error processing CSV: {str(e)}"

# Gradio interface
iface = gr.Interface(
    fn=ask_csv_question,
    inputs=[
        gr.File(label="Upload CSV File", file_types=[".csv"]),
        gr.Textbox(label="Ask a Question About the CSV (e.g. total sales, aver
    ],
    outputs="text",
    title="🇮🇹 Chat with CSV",
    description="Upload a CSV file and ask questions about your data in natura
)

# Launch the CSV chat app
iface.launch()

```