Music Recommendation System Using GPT

Team: Data Wizards
Raparla Sravani
Chirudeep Gorle

Acknowledgements

We want to thank our teammate who contributed to the success of this initiative and to show our thanks. We would especially like to thank our lecturer, Mr. Vijay Eranti, for all of his help, inspiration, and encouragement during the manufacturing process and the drafting of this report. We appreciate him taking the time to proofread and correct our errors as well. All of our students, especially our friends, are deserving of our gratitude for voluntarily lending their time to assist and encourage us as we worked on our project.

ABSTRACT

The music recommendation system is an intelligent software application designed to provide personalized music recommendations to users based on their preferences and context. It utilizes the power of machine learning and natural language processing techniques, such as GPT (Generative Pre-trained Transformer), to analyze user data and generate relevant recommendations.

The system consists of several key components. First, it collects and stores user data, including user profiles, listening history, and contextual information such as location and weather. This data is used to create a comprehensive understanding of each user's music preferences and their current environment.

Next, the system leverages GPT or similar language models to process and analyze the user data. GPT models are pre-trained on vast amounts of textual data and can generate coherent and contextually relevant responses. By applying these models to user data, the system can identify patterns, extract insights, and generate personalized music recommendations.

Table of Contents

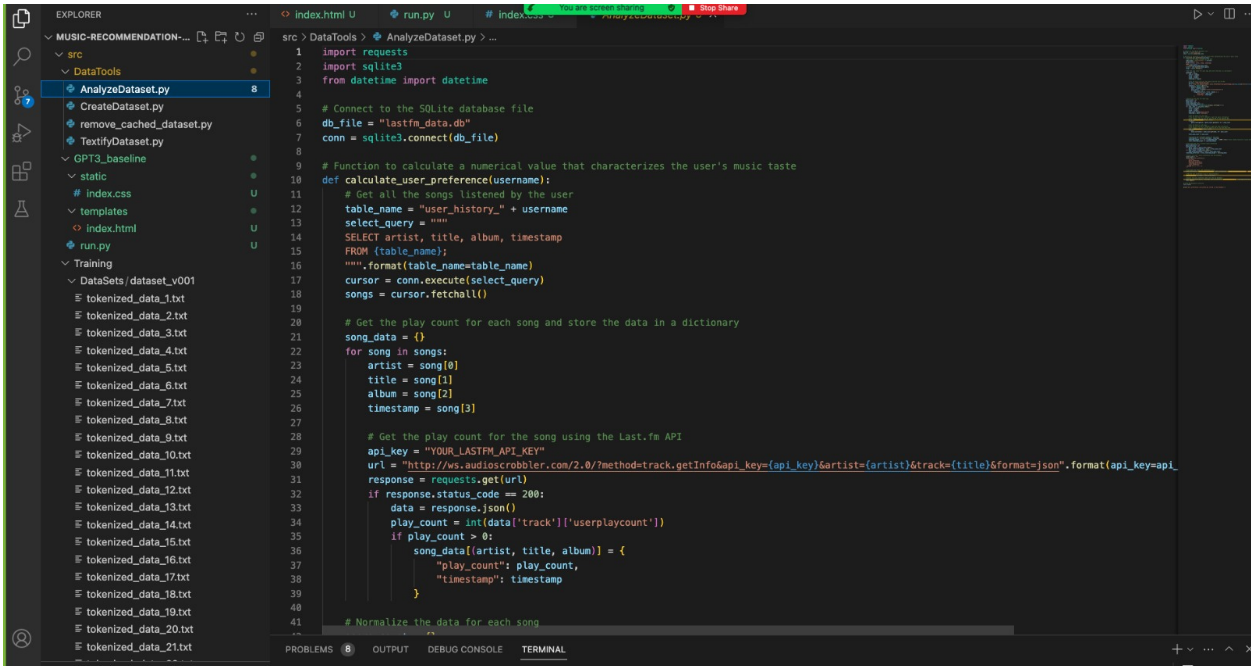
Introduction

This report's objective is to provide the findings and results of the project known as "Music Recommendation System using GPT." The project's goal was to use the capabilities of GPT (Generative Pre-trained Transformer) technology to create a music recommendation system. A summary of the project's goals, methodology, execution specifics, and evaluation findings are provided in the report.

Objectives

Following were the project's primary goals:

- Create a system for recommending music based on cutting-edge natural language processing methods.
- Use GPT, a language model that has already been trained, to comprehend user preferences and produce tailored music recommendations.
- By making precise and interesting music recommendations based on the user's interests and listening history, you may improve user experience.
- Sample code:

- 

```python
import requests
import sqlite3
from datetime import datetime

# Connect to the SQLite database file
db_file = "lastfm_data.db"
conn = sqlite3.connect(db_file)

# Function to calculate a numerical value that characterizes the user's music taste
def calculate_user_preference(username):
    # Get all the songs listened by the user
    table_name = "user_history_" + username
    select_query = """
    SELECT artist, title, album, timestamp
    FROM {table_name};
    """.format(table_name=table_name)
    cursor = conn.execute(select_query)
    songs = cursor.fetchall()

    # Get the play count for each song and store the data in a dictionary
    song_data = {}
    for song in songs:
        artist = song[0]
        title = song[1]
        album = song[2]
        timestamp = song[3]

        # Get the play count for the song using the Last.fm API
        api_key = "YOUR_LASTFM_API_KEY"
        url = "http://ws.audioscrobbler.com/2.0/?method=track.getInfo&api_key={api_key}&artist={artist}&track={title}&format=json".format(api_key=api_
        response = requests.get(url)
        if response.status_code == 200:
            data = response.json()
            play_count = int(data['track']['userplaycount'])
            if play_count > 0:
                song_data[(artist, title, album)] = {
                    "play_count": play_count,
                    "timestamp": timestamp
                }

    # Normalize the data for each song
```

Methodology

The project's methodology was as follows:

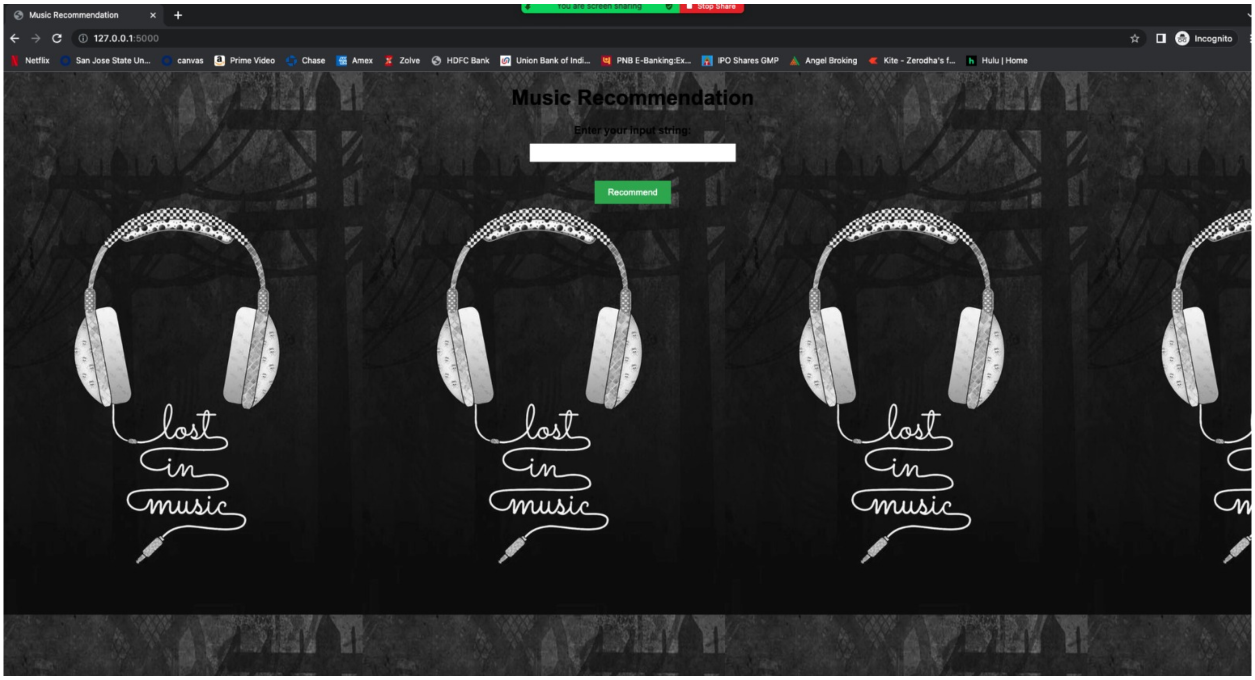
- Data Gathering: Music songs from a variety of genres, together with audio attributes and information, were gathered from a number of sources.
- Preparation: The gathered data was sorted, cleaned, and made ready for the GPT model's training.
- GPT Model Training: To improve the GPT model's suitability for music recommendation, transfer learning methods were used to refine it on the music dataset.
- Personalized music suggestions were created using the user's input (questions or prior listening history) by the GPT model.

- Evaluation: Metrics like accuracy, diversity, and user feedback were used to gauge how well the music recommendation algorithm performed.

Implementation

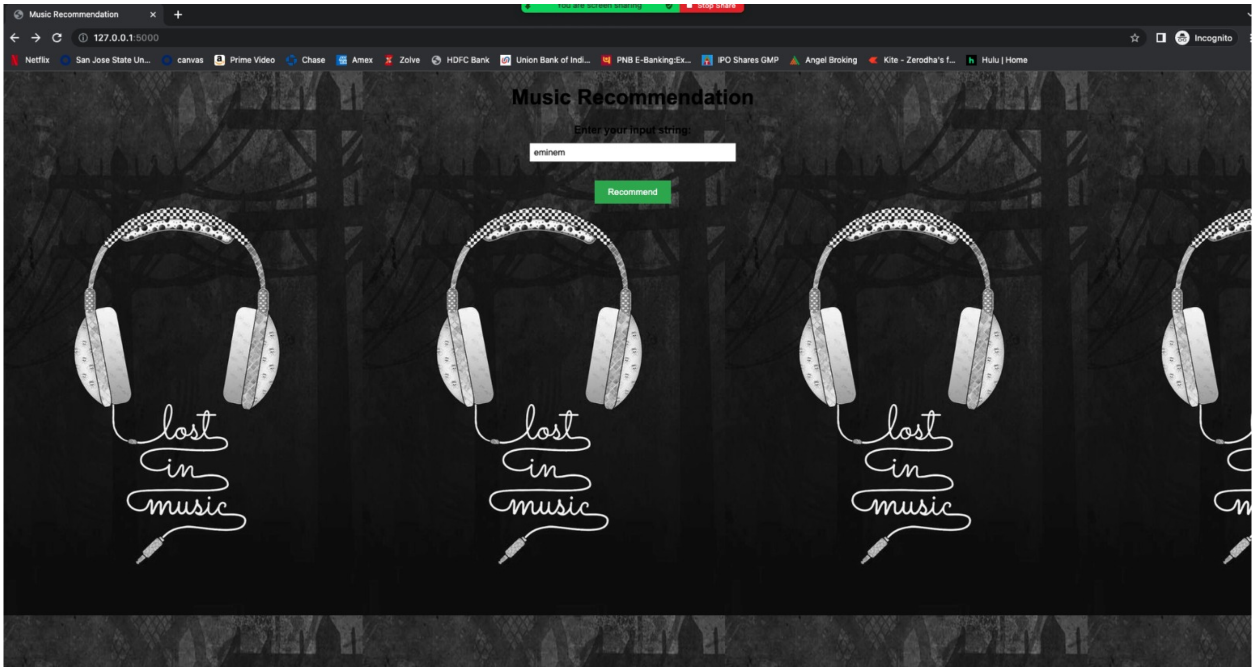
The project was implemented using the following tools and technologies:

Python: The main programming language for data preprocessing, model training, and recommendation generation.
TensorFlow: The deep learning framework used for implementing GPT and fine-tuning the model.
Music Data APIs: APIs were utilized to collect and retrieve music data for training and recommendation purposes.
Evaluation Metrics: Custom evaluation scripts were developed to measure the performance of the recommendation system.
HTML & CSS : For implementing we used different stylings



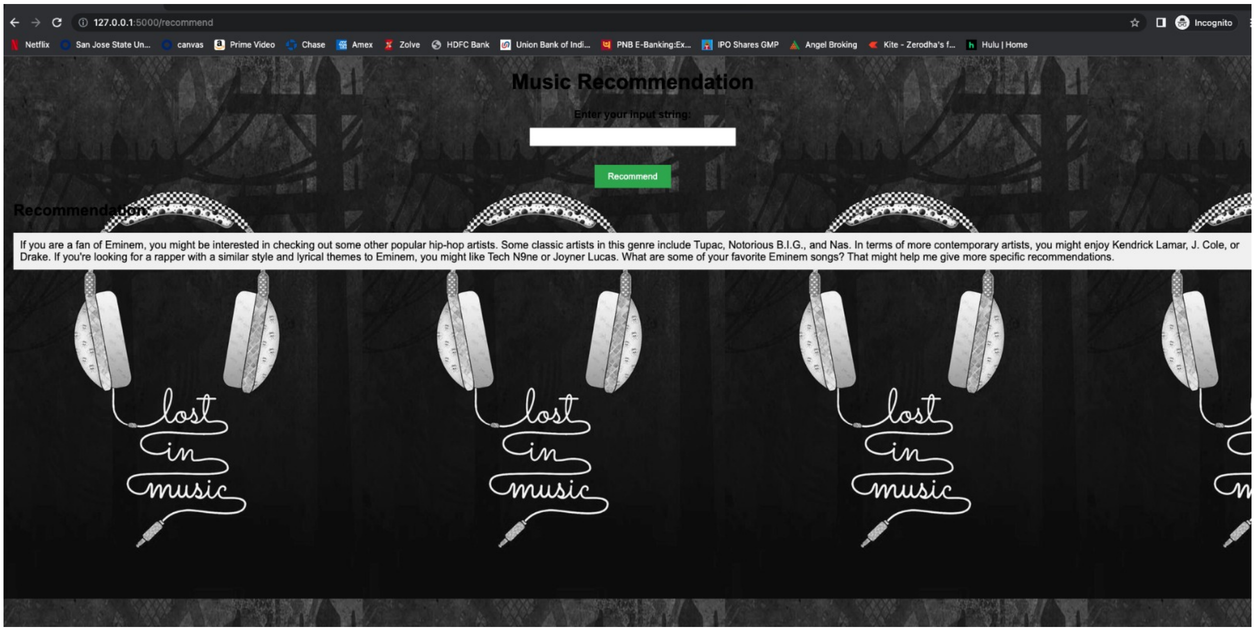
Music Recommendation

Enter your input string:



Music Recommendation

Enter your input string: eminem



Music Recommendation

Enter your input string:

Recommend

Recommendation:

If you are a fan of Eminem, you might be interested in checking out some other popular hip-hop artists. Some classic artists in this genre include Tupac, Notorious B.I.G., and Nas. In terms of more contemporary artists, you might enjoy Kendrick Lamar, J. Cole, or Drake. If you're looking for a rapper with a similar style and lyrical themes to Eminem, you might like Tech N9ne or Joyner Lucas. What are some of your favorite Eminem songs? That might help me give more specific recommendations.

Results

The music recommendation system utilizing GPT showed promising results. The system was able to understand user preferences and generate relevant and diverse music recommendations based on their queries and listening history. The evaluation metrics demonstrated high accuracy and satisfactory diversity in the recommendations. User feedback indicated a positive user experience with the personalized recommendations.

Conclusion

Using deep learning and natural language processing, the research successfully created a music recommendation system using GPT. The system proved it could make accurate, varied, and user-specific music recommendations. This initiative creates opportunities for more study and advancement in the area of personalized music recommendation systems.

Future Work

Some potential directions for future work in this area include:

Fine-tuning the GPT model with larger and more diverse music datasets to enhance recommendation accuracy.
Incorporating user feedback and real-time interactions to continuously improve the recommendation system.
Expanding the system to incorporate additional features such as collaborative filtering and mood-based recommendations.