Music Recommendation System

USING PYTHON

Sooram Karthik | ML using Python | 12-03-2024

Objective:

- The objective of this project is to recommend songs based on user choice using a machine learning model that analyzes the lyrics of songs.

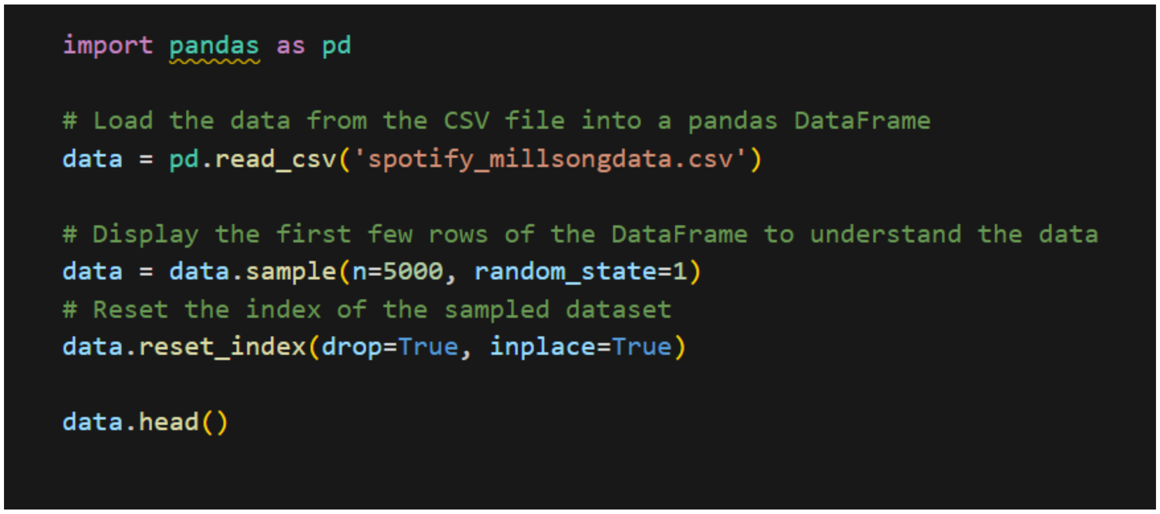
Dataset:

- The dataset used for this project contains information about various songs, including the artist, song title, lyrics, and a link to the song's information. It has been preprocessed to remove null values and unnecessary columns.

Steps to Design Model:

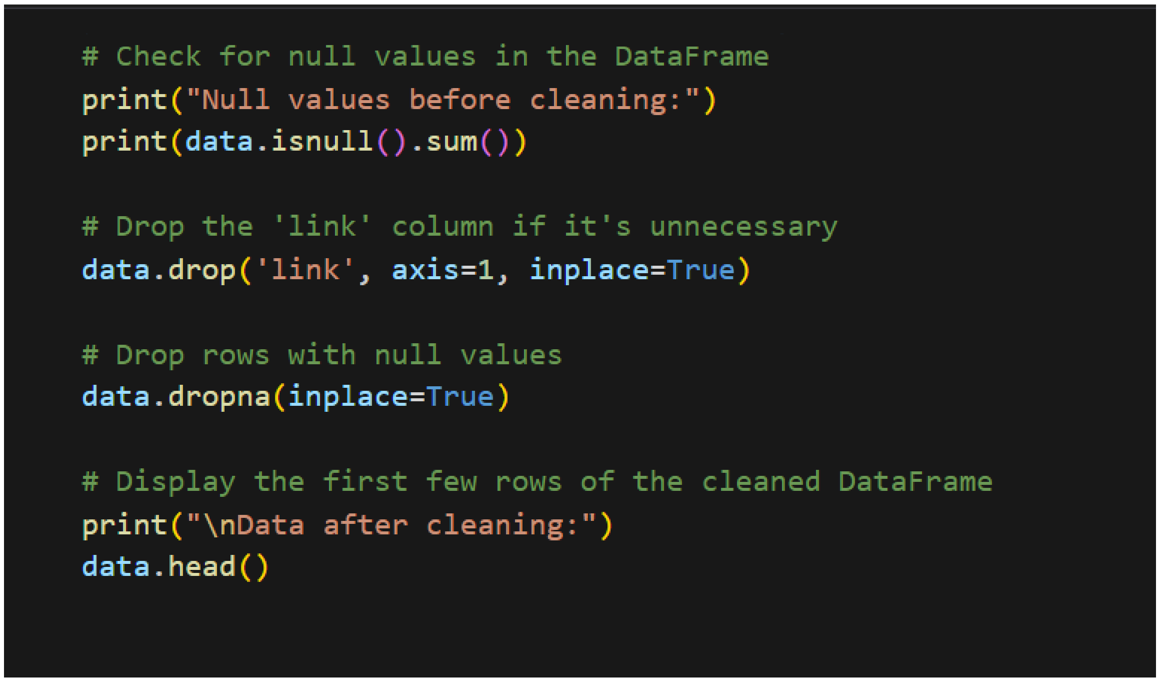
Data Loading and Cleaning:

- Load the dataset from the CSV file into a pandas DataFrame.
- Sample 5000 rows for analysis.
- Reset the index of the sampled dataset.



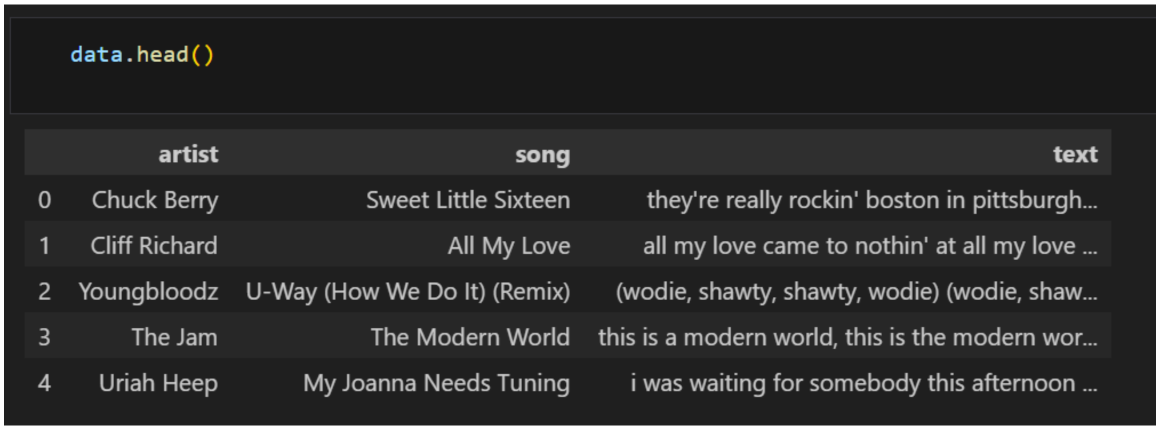
```python
import pandas as pd

# Load the data from the CSV file into a pandas DataFrame
data = pd.read_csv('spotify_millsongdata.csv')

# Display the first few rows of the DataFrame to understand the data
data = data.sample(n=5000, random_state=1)
# Reset the index of the sampled dataset
data.reset_index(drop=True, inplace=True)

data.head()
```

- Remove unnecessary columns like 'link'.
- Remove rows with null values.
- Clean the 'text' column by converting to lowercase and replacing special characters.



```python
# Check for null values in the DataFrame
print("Null values before cleaning:")
print(data.isnull().sum())

# Drop the 'link' column if it's unnecessary
data.drop('link', axis=1, inplace=True)

# Drop rows with null values
data.dropna(inplace=True)

# Display the first few rows of the cleaned DataFrame
print("\nData after cleaning:")
data.head()
```

```python
data['text'] = data['text'].str.lower().replace(r'^\w\s', '
').replace(r'\n', ' ', regex = True).replace(r'\r','
',regex=True)
```

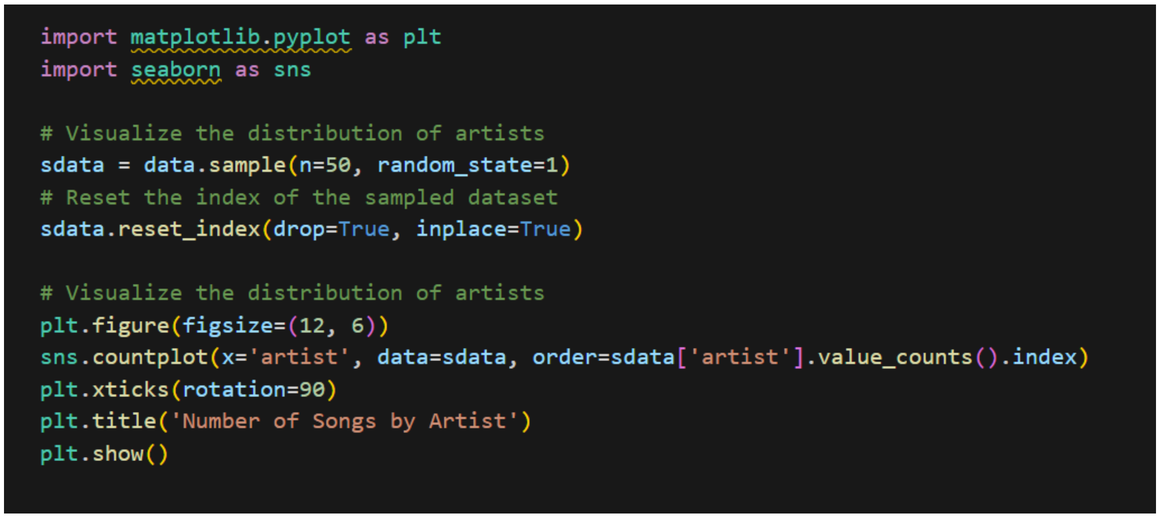


```python
data.head()
```

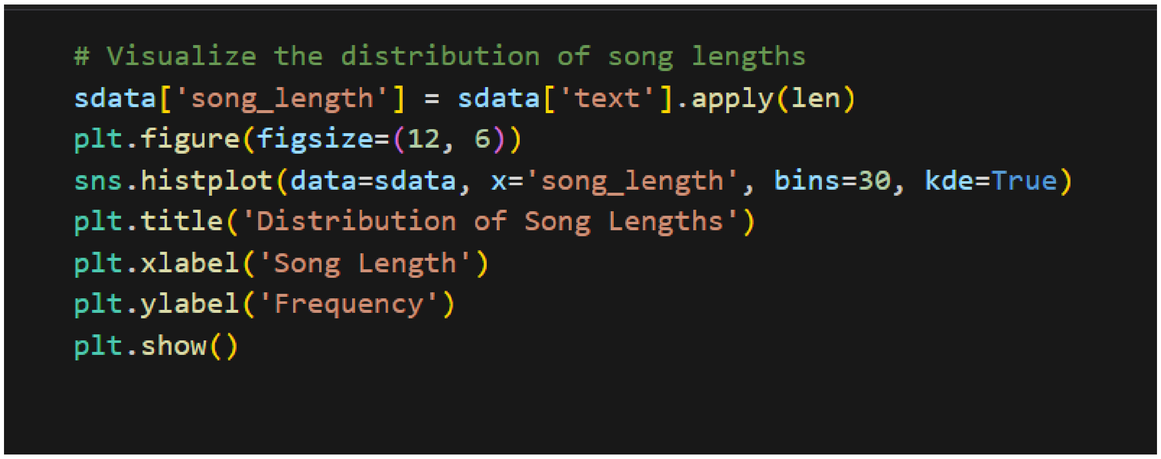
```
         artist                           song                                              text
0   Chuck Berry             Sweet Little Sixteen    they're really rockin' boston in pittsburgh...
1  Cliff Richard                     All My Love    all my love came to nothin' at all my love ...
2   Youngbloodz  U-Way (How We Do It) (Remix)    (wodie, shawty, shawty, wodie) (wodie, shaw...
3       The Jam                 The Modern World    this is a modern world, this is the modern wor...
4    Uriah Heep           My Joanna Needs Tuning    i was waiting for somebody this afternoon ...
```

Data Visualization:

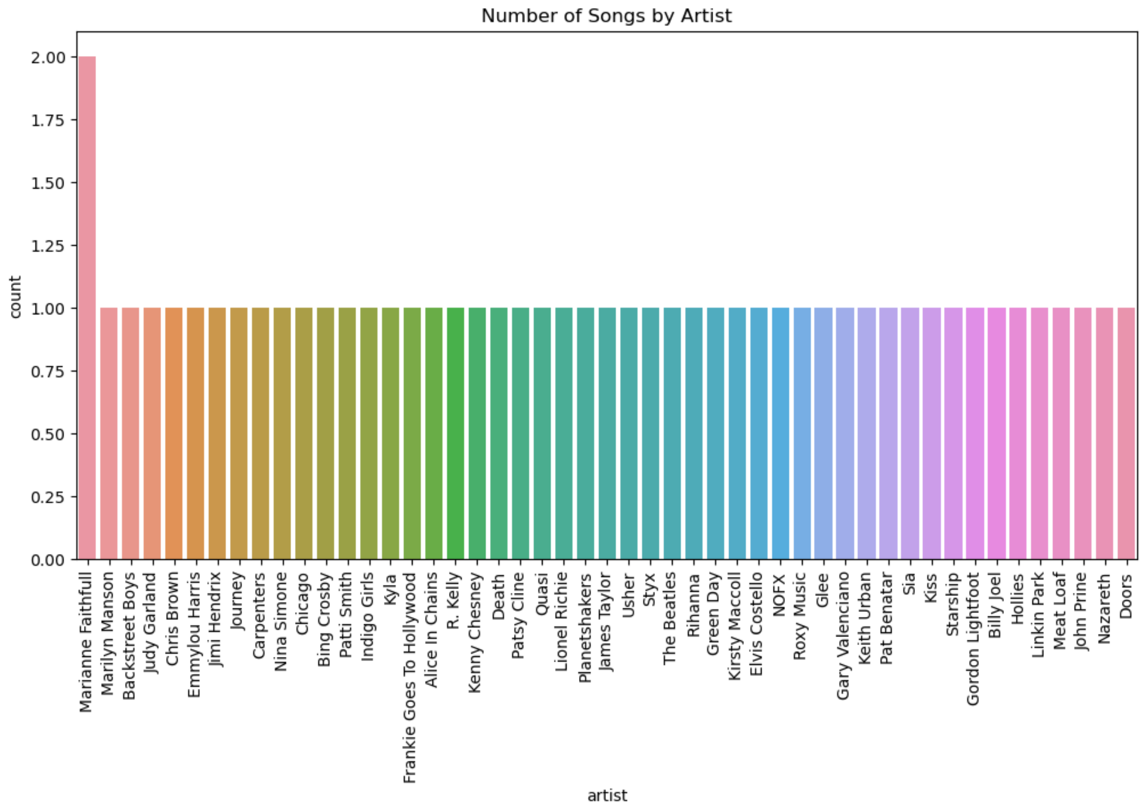
- Visualize the distribution of artists.
- Visualize the distribution of song lengths.

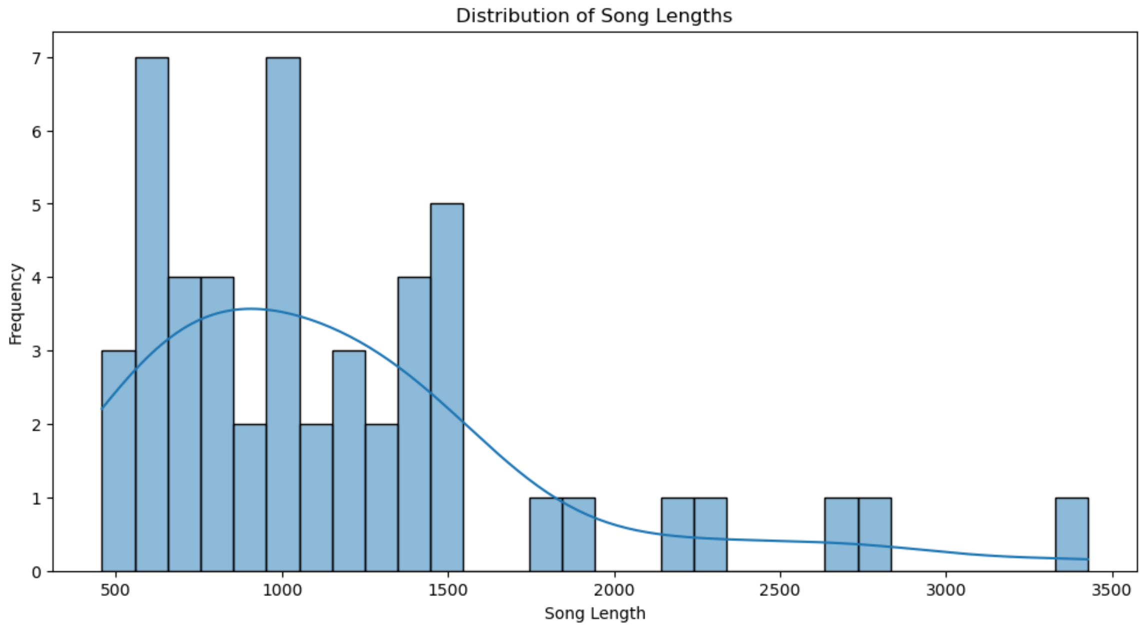


```python
import matplotlib.pyplot as plt
import seaborn as sns

# Visualize the distribution of artists
sdata = data.sample(n=50, random_state=1)
# Reset the index of the sampled dataset
sdata.reset_index(drop=True, inplace=True)

# Visualize the distribution of artists
plt.figure(figsize=(12, 6))
sns.countplot(x='artist', data=sdata, order=sdata['artist'].value_counts().index)
plt.xticks(rotation=90)
plt.title('Number of Songs by Artist')
plt.show()
```



```python
# Visualize the distribution of song lengths
sdata['song_length'] = sdata['text'].apply(len)
plt.figure(figsize=(12, 6))
sns.histplot(data=sdata, x='song_length', bins=30, kde=True)
plt.title('Distribution of Song Lengths')
plt.xlabel('Song Length')
plt.ylabel('Frequency')
plt.show()
```


Number of Songs by Artist
count
artist
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Marianne Faithfull
Marilyn Manson
Backstreet Boys
Judy Garland
Chris Brown
Emmylou Harris
Jimi Hendrix
Journey
Carpenters
Nina Simone
Chicago
Bing Crosby
Patti Smith
Indigo Girls
Kyla
Frankie Goes To Hollywood
Alice In Chains
R. Kelly
Kenny Chesney
Death
Patsy Cline
Quasi
Lionel Richie
Planetshakers
James Taylor
Usher
Styx
The Beatles
Rihanna
Green Day
Kirsty Maccoll
Elvis Costello
NOFX
Roxy Music
Glee
Gary Valenciano
Keith Urban
Pat Benatar
Sia
Kiss
Starship
Gordon Lightfoot
Billy Joel
Hollies
Linkin Park
Meat Loaf
John Prine
Nazareth
Doors


Distribution of Song Lengths
Frequency
Song Length
0
1
2
3
4
5
6
7
500
1000
1500
2000
2500
3000
3500

Model Building:

- Use TF-IDF vectorization to convert lyrics text into numerical form.
- Calculate the cosine similarity matrix to measure similarity between songs based on lyrics.

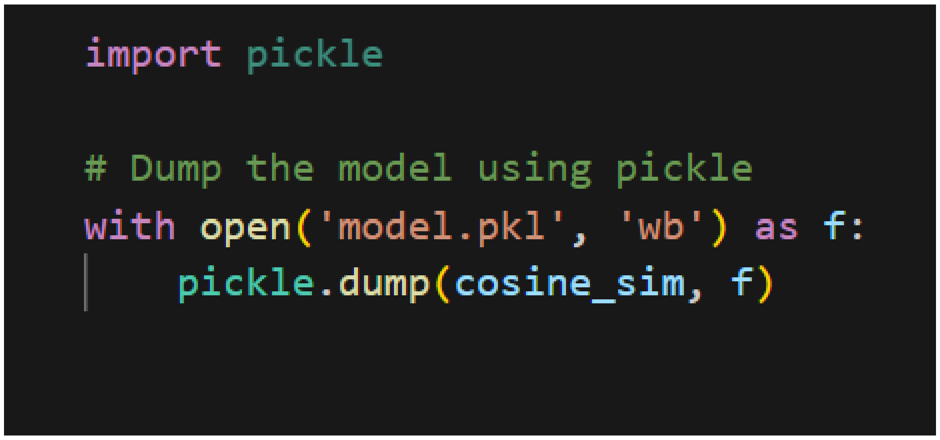
Recommendation Function:

- Define a function to recommend songs based on the selected song.

```python
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

# Create a TF-IDF vectorizer
tfidf_vectorizer = TfidfVectorizer(stop_words='english')

# Fit and transform the lyrics text
tfidf_matrix = tfidf_vectorizer.fit_transform(data['text'])

# Calculate the cosine similarity matrix
cosine_sim = cosine_similarity(tfidf_matrix, tfidf_matrix)

# Function to recommend songs based on the selected song
def recommend_songs(selected_song, cosine_sim=cosine_sim):
    # Get the index of the selected song
    idx = data[data['song'] == selected_song].index[0]

    # Get the cosine similarity scores of the selected song with all other
songs
    sim_scores = list(cosine_sim[idx])

    # Sort the songs based on the similarity scores
    sim_scores_sorted = sorted(range(len(sim_scores)), key=lambda i:
sim_scores[i], reverse=True)

    # Get the top 20 most similar songs
    top_20_indices = sim_scores_sorted[1:21]

    # Return the top 20 recommended songs
    return data['song'].iloc[top_20_indices].tolist()
```

Model Persistence:

- Dump the cosine similarity matrix model using pickle.

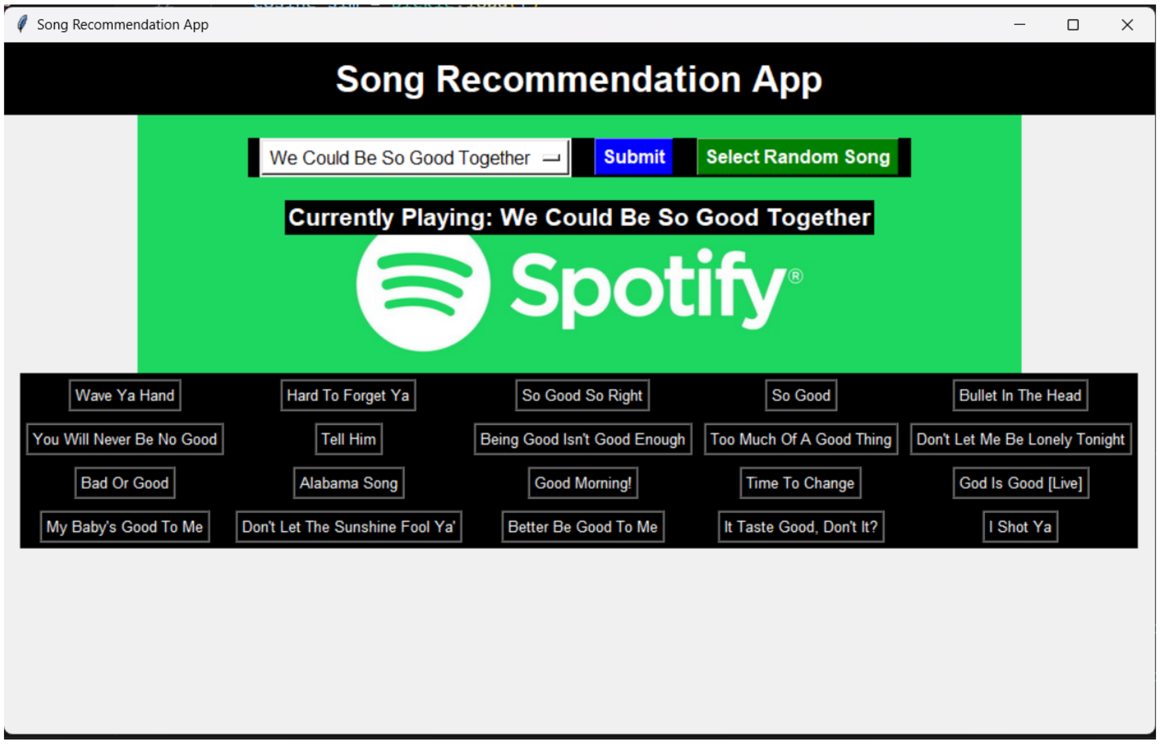


```python
import pickle

# Dump the model using pickle
with open('model.pkl', 'wb') as f:
    pickle.dump(cosine_sim, f)
```

Application:

The application is a Song Recommendation App developed using Tkinter. It allows users to:

- Select a song from a dropdown menu or choose a random song.
- Display the currently playing song.
- Recommend 20 other songs based on the selected song's lyrics similarity.



Song Recommendation App

Song Recommendation App

We Could Be So Good Together

Currently Playing: We Could Be So Good Together

Wave Ya Hand	Hard To Forget Ya	So Good So Right	So Good	Bullet In The Head
You Will Never Be No Good	Tell Him	Being Good Isn't Good Enough	Too Much Of A Good Thing	Don't Let Me Be Lonely Tonight
Bad Or Good	Alabama Song	Good Morning!	Time To Change	God Is Good [Live]
My Baby's Good To Me	Don't Let The Sunshine Fool Ya'	Better Be Good To Me	It Taste Good, Don't It?	I Shot Ya

Drawbacks and Limitations:

- The recommendation is solely based on the lyrics similarity and does not consider other factors such as genre or popularity.
- The dataset size may limit the diversity and accuracy of recommendations.

Conclusion:

In conclusion, this project demonstrates the use of machine learning and natural language processing techniques to recommend songs based on user choice. While the model provides a novel way to discover new music, its effectiveness can be further improved by incorporating additional features and a larger dataset