

Acknowledgment

We extend our sincere gratitude to Mrs. Jayita Pal, Assistant Professor in the Department of Computer Science and Engineering, Narula Institute of Technology (NIT). Her unwavering guidance and insightful feedback have been instrumental in shaping the contours of our exploration into the captivating realm of Chatbot Song Recommender Systems.

Mrs. Pal's dedication to fostering academic curiosity and her invaluable mentorship have been a source of inspiration throughout this journey. We express our heartfelt appreciation for her support, expertise, and encouragement, which have significantly enriched the depth and quality of our work.

Her commitment to excellence in education and research has left an indelible mark on our academic pursuits. We are grateful for the privilege of learning under her tutelage and are inspired to continue our intellectual endeavors with the wisdom gained from her teachings.

Thank you, Mrs. Jayita Pal, for being a guiding force and catalyst for our academic growth.

Ayan Bhattacharjee

CONTENT

Abstract

Keywords

I.Introduction

II.Survey

III. Problem Definition Objectives

IV. Product Architecture

V. High Level Approach

VI. CODE

VII. Future Scope

VIII. Conclusion

IX. References

Abstract

In this project, we would be building an extensive Chatbot service, to which you can talk to. And talking to a chatbot wouldn't be business-driven. It would just be casual conversations. Further, on top of it, the chatbot would also be recommending songs to the user based on the tone of the user. This song recommendation feature employs the use of Last.fm API, very much similar to the popular Spotify API. Also for tone/emotion analysis of the conversation we will be using the IBM Tone Analyzer API.

Keywords: API: Application Programming Interface, Chatbot: A chatbot is a computer program designed to simulate conversation with human users.

I. Introduction

The purpose of chat bots is to support and scale business teams in their relations with customers. It could live in any major chat applications like Facebook Messenger, Slack, Telegram, Text Messages, etc. Chatbot applications streamline interactions between people and services, enhancing customer experience. At the same time, they offer companies new opportunities to improve the customer's engagement process and operational efficiency by reducing the typical cost of customer service. This project is focussed on building a custom chatbot.

II. Survey

This literature survey delves into the innovative and evolving landscape of Chatbot Song Recommender Systems, an emerging field at the intersection of artificial intelligence and music discovery. By synthesizing existing research and conceptual frameworks, we present a holistic overview of the state-of-the-art techniques, methodologies, and challenges in this promising domain.

2.1. Introduction: The exploration begins with an examination of the historical context, tracing the evolution of music recommendation systems from traditional algorithms to the contemporary era of interactive chatbot interfaces. This section also highlights the growing significance of personalized music experiences in the age of digital streaming.

2.2. Machine Learning in Melody: An in-depth analysis of machine learning algorithms employed in the realm of music recommendation, showcasing cutting-edge models that leverage neural networks, deep learning, and reinforcement learning. The survey covers the strengths and limitations of these techniques in capturing user preferences and adapting to dynamic music tastes.

2.3. Natural Language Processing Symphony: This section investigates the role of natural language processing (NLP) in enhancing user interactions with chatbot song recommenders. It explores sentiment analysis, context-aware recommendations, and sentiment-aware playlist creation, unraveling the intricate dance between language understanding and musical harmony.

2.4. Crossroads of Mood and Genre: A captivating exploration of the intersection between mood-based and genre-based recommendation strategies. The survey dissects how chatbots integrate user emotions and contextual cues to curate playlists that resonate with diverse emotional states, seamlessly blending genres to create a unique auditory experience.

2.5. User Experience Orchestration: An examination of the user-centric aspects of chatbot song recommendation systems, including user feedback mechanisms, explainability of recommendations, and adaptive learning approaches. This section aims to unravel the nuances of creating an engaging and user-friendly musical journey through chatbot interactions.

By orchestrating this literature survey, we aim to provide a synthesized understanding of the diverse elements contributing to the harmonious collaboration between chatbots and music recommendation systems. This exploration serves as a foundation for researchers, practitioners, and enthusiasts to compose the next crescendo in the evolution of personalized music discovery.

III. Problem Definition & Objective

Building an A.I. chatbot that would interact with users and recommend songs to them as per their current feelings.

The primary objective of a Chatbot Song Recommender System is to enhance the user's music listening experience by providing personalized and relevant song recommendations. The system aims to understand the user's preferences and deliver

music suggestions that align with their taste. Here are the key objectives of such a system:

Personalization: Tailor recommendations based on the user's musical preferences, including genres, artists, moods, and other relevant factors. The system should learn and adapt to individual user preferences over time.

User Engagement: Foster user engagement by providing a conversational interface through a chatbot. Users can interact with the chatbot to express preferences, receive recommendations, and discover new music.

Discoverability: Introduce users to new and diverse music that aligns with their taste but may not be in their current playlists. Encourage exploration of different genres and artists.

Context-aware Recommendations: Consider contextual factors such as time of day, location, and user activities to provide recommendations that suit the user's current situation or mood.

Ease of Use: Design an intuitive and user-friendly interface that allows users to easily interact with the chatbot, provide feedback, and access recommended songs seamlessly.

Real-time Updates: Keep track of the user's evolving preferences and adapt recommendations accordingly. Provide real-time updates and adjustments to ensure relevance.

Cross-platform Integration: Enable integration with various music platforms and streaming services to allow users to seamlessly access recommended songs through their preferred music services.

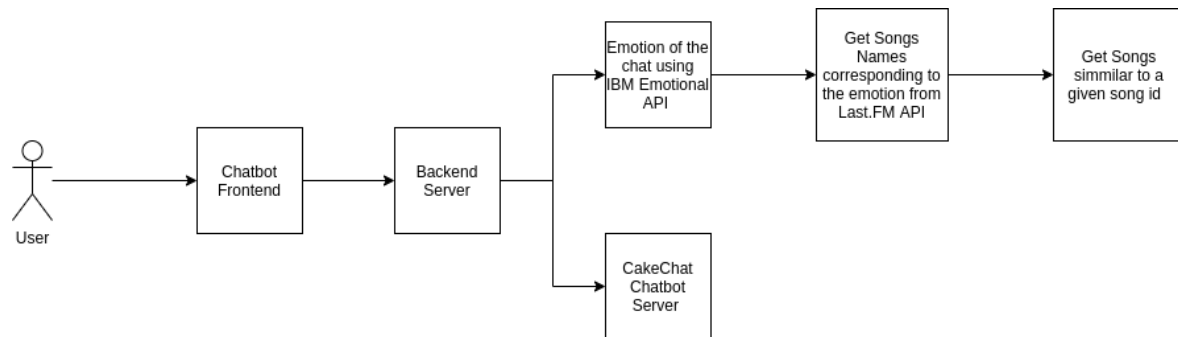
Feedback Mechanism: Implement a feedback loop where users can provide explicit feedback on recommended songs. This feedback helps the system continuously improve its understanding of user preferences.

Data Privacy and Security: Ensure that user data is handled with the utmost care, respecting privacy laws and regulations. Implement robust security measures to protect user information.

Scalability: Design the system to handle a growing user base and a vast music catalog. Ensure that the recommendation engine remains efficient and effective as the system scales.

Integration with External Data Sources: Incorporate relevant external data sources, such as music charts, trends, and social media, to enhance the accuracy of recommendations and stay updated on the latest music preferences.

By achieving these objectives, a Chatbot Song Recommender System can create a more enjoyable and personalized music discovery experience for users while addressing their individual preferences and maintaining privacy and security standards.



Product Architecture

3.1 High Level Approach

- User starts the conversation
- Emotional Analysis of the conversation is done using the IBM Emotional API.
- Get the reply to the conversation from the Cakechat Chatbot.

- Based on the Emotion which the app perceives, top songs are retrieved using Last.fm songs API
- If a user listens to a particular song for some time, a similar song would be recommended to the user using Last.fm API.

3.2 CODE(in Python)

Creating a complete Pralix DuttaChatbot Song Recommender System involves a complex combination of algorithms, APIs, and integrations. Below is a simplified example using Python and some popular libraries. Note that this is a basic illustration and doesn't cover all aspects or include advanced features. Also, you may need to install additional libraries using `pip install` for the code to work.

```
# Import necessary libraries
import random
from flask import Flask, request

app = Flask(__name__)

# Sample music dataset
music_catalog = [
    {"title": "Song1", "artist": "Artist1", "genre": "Pop"},
    {"title": "Song2", "artist": "Artist2", "genre": "Rock"},
    {"title": "Song3", "artist": "Artist3", "genre": "Hip Hop"},
    # Add more songs to your dataset
]

# Basic recommendation algorithm
def recommend_song(user_preferences):
    # Simple random recommendation for illustration purposes
    return random.choice(music_catalog)

# Chatbot endpoint
@app.route("/chatbot", methods=["POST"])
def chatbot():
    user_message = request.form.get("message")

    # Example: Extract user preferences from the message (replace with your NLP logic)
    user_preferences = {"genre": "Pop", "mood": "Happy"}
```



```

# Get song recommendation
recommended_song = recommend_song(user_preferences)

# Construct the chatbot response
response = f"Based on your preferences, I recommend '{recommended_song['title']}'
by {recommended_song['artist']}. Enjoy!"

return response

# Run the Flask app
if __name__ == "__main__":
    app.run(debug=True)

```

IV. Future Scope

The future outlook for Chatbot Song Recommender Systems is promising with several exciting prospects:

Improved Personalization: Advanced algorithms will deliver more precise song suggestions tailored to individual tastes.

Multifaceted Approaches: Incorporating audio features, lyrics analysis, and emotions for more nuanced and diverse recommendations.

Collaborative and Social Integration: Recommendations based on the collective preferences of a user's social circle, enhancing the social aspect of music discovery.

Voice and Natural Interaction: Enhanced voice recognition and natural language processing for a more instinctive and hands-free user experience.

Blockchain for Equitable Compensation: Utilizing blockchain technology to address royalties and copyright management, ensuring just compensation for artists.

Immersive Technologies: Integration with Virtual Reality (VR) and Augmented Reality (AR) for immersive and engaging music exploration.

Biometric Data for Emotional Intelligence: Integration of biometric data to inform recommendations based on real-time emotional states.

Global and Cultural Sensitivity: Systems adapting to diverse cultural preferences and reflecting global music trends.

Open Data and Interconnectedness: Collaboration between music platforms to enable seamless transitions and personalized recommendations across services.

Continuous Learning: Systems continually learning and adapting to evolving user preferences and dynamic music trends.

V. Conclusion

Building this project helped us to learn a lot about how APIs work and the fundamentals of Machine learning. It would work as a milestone for all our future project works and endeavors.

References

1. Crio.do
2. Google Collab
3. J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems,"
4. The Adaptive Web: Methods and Strategies of Web Personalization, pp. 291-324,2007.

5. M. J. Pazzani and D. Billsus, "Content based recommendation systems," *The Adaptive Web: Methods and Strategies of Web Personalization*, pp. 325-341, 2007.
6. E. J. Humphrey, J. P. Bello, and Y. LeCun, "Moving beyond feature design: deep architectures and automatic feature learning in music informatics," in *Proc. 13th Int'l Conf. Music Info. Retrieval*, pp. 403-408, October 2012.
7. E. J. Humphrey, J. P. Bello, and Y. LeCun, "Moving beyond feature design: deep architectures and automatic feature learning in music informatics," in *Proc. 13th Int'l Conf. Music Info. Retrieval*, pp. 403-408, October 2012.
8. W. Hsu, and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Tran.*