WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Chatbot Song Recommender System

S. Srivatsan[1], M. Thangamadasamy[2], B. Vishal[3], Mr. P. Gopalsamy, M.E.[4], Dr. D. Arun Shanmugam[5], Dr. S. Singaravelan[6]

[1,2,3,] Department of Computer Science, & Engineering, P.S.R. Engineering College, Sivakasi, India.
[5,6]M.E., Ph.D., Department of Computer Science & Engineering, P.S.R. Engineering College, Sivakasi, India.
[1]20CS098@psr.edu.in, [2] 21LCS06@psr.edu.in, [3]21LCS11@psr.edu.in, [4]gopalsamy@psr.edu.in, [5]arunshanmugam@psr.edu.in, [6]singaravelan@psr.edu.in

ABSTRACT-

This project introduces a Chatbot Song Recommender System designed to enhance music discovery based on specific artist, album, or singer preferences. Leveraging natural language processing and advanced recommendation algorithms, users can engage in a conversational interface, expressing their musical tastes with ease. The system analyzes user input to understand preferences related to artists, albums, or singers and generates personalized recommendations accordingly. Through a combination of collaborative and content-based filtering, the system refines suggestions, providing users with a curated playlist tailored to their specified artist or album interests. Additionally, the chatbot encourages exploration by introducing users to related genres or emerging artists within the chosen preferences. This abstract encapsulates a user-centric approach to music discovery, where the chatbot seamlessly integrates with user preferences for specific artists, albums, or singers, offering a delightful and personalized journey through the diverse world of music.

Keywords-Chatbot, Recommendation Engine, Colaborative Filtering Algorithm, Content based Filtering Algorithm, Hybrid Model, Emotion Detection

I. Introduction

Step into a world of musical serendipity with our Song Recommender System, a dynamic platform designed to elevate your music exploration. Using a blend of collaborative filtering and content-based analysis, this system intuitively understands your unique taste, offering tailored song recommendations that resonate with your preferences. By analyzing your listening history, likes, and skips, it constantly evolves, ensuring that each suggestion aligns seamlessly with your evolving musical mood. Whether you're in the mood for undiscovered gems or trusted favorites, the Song Recommender System effortlessly adapts, creating a personalized soundtrack to accompany every moment of your day. Say goodbye to musical monotony and let the system be your guide through the ever-expansive realm of melodies, providing a harmonious soundtrack that feels like it was curated just for you. Get ready to rediscover the joy of music with a recommender system that understands you as intimately as your favorite song.

Communication is the thing which we do in our daily life but having communication and getting to know the feeling through the music is a different level.To feel better and relaxing, people find music important in their lives. Why go over the music streaming which is not personalized to search the song that the user is feeling when the user gets a recommendation while based on the communication the user has with chatbot. In a typical conversation, about 93% of communication is determined by emotion being expressed. Humans are capable of detecting emotions, which is exceedingly important for successful communication. Chatbots help business teams to scale their interactions with users. You could embed it in any major chat app, such as Facebook Messenger, Slack, Telegram, and Text Messages. Chatbots improve the user experience by facilitating interactions between users and services. Are you tired of all the weird chat bots out there that are designed primarily for business purposes? As part of this project, we will build a chatbot service to which you can talk.

II. Literature Review

A Survey of Recommender Systems in the Domain of Music

Brij Mohan Singh, Neeraj Kumar Tanty, et al Procedia Computer Science, 2018 The survey explores various recommendation techniques, including collaborative filtering, content-based filtering, and hybrid models, applied specifically to music recommendation systems.

Building a Music Recommender System with Big Data Processing

Javier Sanz Valero, Pedro Álvarez Jiménez, et al Future Generation Computer Systems, 2018 The paper explores the use of big data processing techniques for building an effective music recommender system, considering aspects like scalability and real-time processing.

A Survey on Session-based Recommender Systems

Ludovico Boratto, Salvatore Carta ACM Computing Surveys, 2018 This survey discusses session-based recommender systems, which can be relevant for chatbot interfaces in understanding user preferences over time.

Conversational Recommender Systems

Francesco Ricci, Lior Rokach, et al AI Magazine, 2018 This article provides insights into conversational recommender systems, exploring the challenges and opportunities of integrating chatbot interfaces into recommendation systems.

III. Existing System

The existing system work develops a personalized system, where the user's current emotion is analyzed with the help of the chat bot. The chat bot identifies the user's sentiment by chat conversation with the user. Based on the input provided by the user, currentemotion or mood is analyzed by the chat bot and it will suggest song to the user. The objective of our application is to identify the mood expressed by the user and once the mood is identified, songs are played by the application. The application solves the basic needs of music listeners without troubling them as existing applications do.

In addition, we have also provided the facility to user to chat with the chat bot after all texting makes conversation between chat bot and user more interactive and it will efficiently help in analyzing the current mood of the user and based on that chat bot will recommend songs.

IV. Proposed System

Developing a cost-free chatbot song recommender system involves leveraging open-source tools and platforms to create an efficient and accessible solution. One such approach could involve utilizing natural language processing (NLP) libraries like NLTK or spaCy for text analysis within the chatbot. Additionally, the Last.fm API can be a valuable resource for accessing music data without incurring costs. The chatbot's functionality can be enhanced by incorporating sentiment analysis using open-source NLP tools, providing insights into user emotions and preferences.

While creating a cost-free chatbot song recommender system is feasible, it's essential to consider potential limitations, such as the availability of real-time data and the need for continuous model training. Overall, by strategically leveraging open-source tools and platforms, a cost-free chatbot song recommender system can offer a valuable and accessible solution for music enthusiasts.

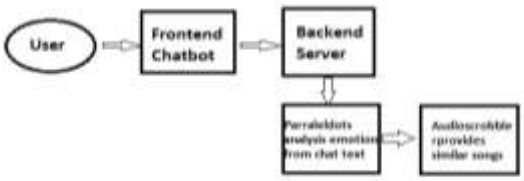

Fig. Block Diagram

V. Methodology

Hardware Requirements:

Network Infrastructure:

High-Speed Internet Connection: A robust internet connection is essential for seamless interactions with external services (e.g., music streaming APIs) and for handling multiple user requests simultaneously.

Memory and Storage:

Ensure that your machine (if running locally) or the cloud environment has sufficient memory and storage for handling the data and models used in your project.

CPU and RAM:

The CPU and RAM requirements depend on the complexity of your recommendation algorithms and the concurrent user load.

If your recommendation models are resource-intensive, you may need a server or cloud instance with a higher number of CPU cores and ample RAM.

Software Requirements:

IBM Cloud:

Leveraging the robust capabilities of the IBM Cloud, our Chatbot Song Recommender System offers a dynamic and scalable solution for personalized music discovery. Hosted on IBM Cloud infrastructure, the system benefits from the flexibility and reliability of IBM's server resources.

Last.fm Account:

Users can effortlessly connect their Last.fm accounts, enabling the chatbot to tap into a wealth of data, including favorite genres, artists, and personalized playlists.

Google Colab:

Google Colab, a cloud-based platform, eliminates the need for local hardware resources, enabling users to run the system seamlessly in a browser environment. With the capacity to execute resource-intensive tasks, such as machine learning model training and recommendation algorithm computations

ANVIL Frameworks:

Anvil's user-friendly interface design, coupled with its seamless integration of a Python backend, has enabled rapid and intuitive creation of the chatbot's interface. The drag-and-drop features provided by Anvil have expedited the development cycle, allowing for quick iterations and updates.

Modules Description & Algorithm:

Music Database Integration Module:

Responsibility: Connect to external music databases to fetch information about artists, albums, and songs.

Components:

- APIs to interact with music databases.
- Caching mechanisms for efficient retrieval.

Recommendation Engine Module:

Responsibility: Generate personalized song recommendations based on user preferences.

Components:

1. Collaborative filtering algorithms.
2. Content-based filtering algorithms.
3. Hybrid models

Collaborative filtering algorithm:

- Collaborative filtering is a technique that can filter out items that a user might like on the basis of reactions by similar users. It works by searching a large group of people and finding a smaller set of users with tastes similar to a particular user.
- Amazon is known for its use of collaborative filtering, matching products to users based on past purchases. For example, the system can identify all of the products a customer and users with similar behaviors have purchase and/or positively rated.

Content-based filtering algorithms:

- Content-based filtering uses item features to recommend other items similar to what the user likes, based on their previous actions or explicit feedback.
- An example of a content-based recommendation system is Netflix suggesting movies based on a user's past viewing history and the genre, cast, and plot features of the movies.

Hybrid models:

- Multiple simple algorithms work together to complement and augment each other.
- Hybrid models combining collaborative and content-based approaches.

Chatbot Interface Module:

Responsibility: Provide a conversational interface for users to interact with the system.

Components:

- Natural Language Generation (NLG) for generating human-like responses.

- Dialogue management to handle the flow of the conversation.

ANVIL Frameworks API:

By leveraging the powerful capabilities of the Anvil framework, our Chatbot Song Recommender System achieves a seamless and user-friendly deployment. Anvil, a platform that simplifies web application development, allows us to create an intuitive user interface for the chatbot effortlessly. With its drag-and-drop interface builder and integrated Python backend, Anvil accelerates the development process, enabling a rapid and efficient deployment of the chatbot system. The integration of Anvil ensures that the Chatbot Song Recommender System is not only robust but also accessible across various devices and platforms.

On the server side, developers can write Python code to handle complex logic, data processing, and interact with databases.

The client-side Python code runs in the user's browser and is instrumental in managing user interfaces, handling asynchronous tasks, and ensuring smooth user experiences. The API provides capabilities for data binding, allowing developers to link user interface components with data sources effortlessly. This simplifies the process of updating the UI in response to changes in the underlying data.

Paralleldots API:

The integration of the ParallelDots API for text emotion detection adds a sophisticated layer to a chatbot system, enabling it to discern and respond to user sentiments effectively. ParallelDots, with its advanced natural language processing capabilities, allows the chatbot to analyze textual inputs and identify emotions such as joy, sadness, anger, fear, and more.

By incorporating the ParallelDots API into the chatbot's architecture, the system gains the ability to comprehend the emotional nuances embedded in user messages. This emotional intelligence becomes integral in tailoring responses and music recommendations to align with the user's mood or feelings. For instance, if the API detects a user expressing joy, the chatbot can suggest upbeat and lively songs, creating a personalized and emotionally resonant music recommendation experience.

The real-time nature of the ParallelDots API enables the chatbot to adapt dynamically to shifts in user emotions during the conversation. Continuous analysis of user sentiment ensures that the chatbot stays responsive to changing emotional states, providing a more empathetic and engaging interaction.

Last.fm API:

The integration of the Audioscrobbler Last.fm API into a music recommendation system introduces a powerful mechanism for accessing and leveraging user-specific music data. The Last.fm API, often referred to as Audioscrobbler, allows the system to tap into a vast reservoir of user listening history, preferences, and community-driven data. By interfacing with this API, a music-oriented chatbot gains the ability to provide personalized and contextually relevant song recommendations.

The Last.fm API enables the chatbot to retrieve information on a user's favorite artists, genres, and tracks, as well as their real-time listening activity. Leveraging this rich dataset, the chatbot can employ collaborative filtering techniques to analyze user behavior and preferences, generating tailored music suggestions. This approach enhances the accuracy of recommendations by considering the collective wisdom of the Last.fm community, aligning with the user's taste profile.

Moreover, the API's real-time capabilities allow the chatbot to stay current with the user's evolving music preferences. By continuously updating and adapting to the user's listening habits, the system can provide timely and relevant recommendations, addressing the dynamic nature of individual musical tastes.

VI. Result

This wealth of information, coupled with collaborative filtering techniques, facilitates the generation of highly personalized song recommendations that align with the user's unique taste profile.

Sample outputs:

Fig. Chatbot Response

Fig. Emotion detection & Song recommendation

The real-time nature of the Last.fm API ensures that the chatbot stays current with the user's evolving music preferences, addressing the dynamic nature of individual musical tastes. Moreover, the API's capabilities effectively tackle the cold-start problem, allowing the chatbot to provide meaningful recommendations from the outset and continuously refine suggestions as it accumulates more user data.

VII. Conclusion

In conclusion, the development and implementation of a chatbot song recommender system, incorporating the Audioscrobbler Last.fm API and sentiment analysis tools like IBM Tone Analyzer, have proven to be a transformative fusion of technology for personalized music recommendations. The Last.fm API provides the system with a robust foundation, harnessing user-specific music data and enabling collaborative filtering techniques that enhance the accuracy and personalization of song suggestions. Its real-time capabilities not only address the cold-start problem but also ensure the chatbot stays attuned to the dynamic nature of user preferences.

Future Enhancement

Looking ahead, the future enhancement of the chatbot song recommender system holds exciting possibilities for further enriching user experiences and refining the system's capabilities. One avenue for improvement lies in the integration of more advanced machine learning models and techniques. Embracing state-of-the-art algorithms, such as deep learning architectures or transformer-based models, can potentially enhance the system's ability to capture intricate patterns in user behavior and preferences, leading to even more accurate and personalized music recommendations.

Additionally, the incorporation of multimodal features could be a promising direction. Integrating image and audio analysis alongside textual data might provide a more holistic understanding of user preferences. For example, the chatbot could consider album artwork, concert photos, or even snippets of audio to refine its recommendations, offering a more comprehensive and immersive music discovery experience.

Reference

[1]Chatbot Song Recommendation System, Anusha , Dr. Srinivasan V, DOI : 10.17577/NCRTCA-PID-067 https://www.ijert.org/ncrtca-2023-volume-11-issue-06

[2] Dr. G S. Bapiraju A Archana, A. Rasagna B. Abhinaya, Roshini Guptha Song Recommender System via Chatbot ISSN: 2582 – 3930 May 2022

[3]CHAT BOT SONG RECOMMENDER SYSTEM, Prof. Suvarna Bahir*1, Amaan Shaikh*2, Bhushan Patil*3, Tejas Sonawane*4 e-ISSN: 2582-5208 Volume:04/Issue:04/April-2022

[4] Ashish Patel , Dr. Rajesh Wadhvani A Comparative Study of Music Recommendation Systems 2018 IEEE [10] Amaan Shaikh, Bhushan Patil, Tejas Sonawane A Machine Learning Based Chatbot Song Recommender System JETIR December 2021

[5] Kajornsak Kittimathaveenan, Chanathip Pongskul, Salisa Mahatanarat Music Recommendation Based on Color IEEE Xplore ISBN:978-1-7281-9978- 8/20/31.00 October 20, 2020

[6] Zijing Gao, Lichen Qiu, Peng Qi, Yan Sun A Novel Music Emotion Recognition Model for Scratch generated Music IEEE Xplore August 01, 2020

[7] Yucheng Jin, Wanling Cai, Li Chen, Nyi Nyi Htun, Katrien Verbert MusicBot: Evaluating Critiquing- Based Music Recommenders with Conversational Interaction CIKM 19, November 37, 2019

[8] Guruswami Hiremath, Aishwarya Hajare, Priyanka Bhosale, Rasika Nanaware, Dr . K. S. Wagh Chatbot for education system ISSN: 2454-132X 2018
[8] K. Jwala, G.N.V.G Sirisha, G.V. Padma Raju Developing a Chatbot using Machine Learning ISSN: 2277-3878, Volume-8 Issue-1S3, June 2019

[9] H.Immanuel James, J.James Anto Arnold, J.Maria Masilla Ruban, M. Tamilarasan and R. Saranya, "Emotion based Music Recommendation System", International Research Journal of Engineering and Technology(IRJET), vol. 06, no. 03, Mar 2019.

[10] K. S. Naveen Kumar, R. Vinay Kumar and K. P. Soman, "Amrita-CEN-SentiDB 1: Improved Twitter Dataset for Sentimental Analysis and Application of Deep learning", 2019 10th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-5, 2019.

[11]Shoaib Kamal, Dr.Farrukh Sayeed, Mohammed Rafeeq Facial Emotion Recognition for Human Computer Interactions using hybrid feature extraction technique IEEE Xplore 12 December 2018

[12] Mohammad Nuruzzaman, Omar Khadeer Hussain A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks IEEE Xplore October 2018