Implementing a Chatbot Music Recommender System Based on User Emotion

Neenumol Mathew
Computer Science and Digital Technologies Department, School of Architecture, Computing and Engineering, UEL, London, UK
u2221274@uel.ac.uk

Nitish Chooramun
Computer Science and Digital Technologies Department, School of Architecture, Computing and Engineering, UEL, London, UK
N.Chooramun@uel.ac.uk

Mhd Saeed Sharif
Computer Science and Digital Technologies Department, School of Architecture, Computing and Engineering, UEL, London, UK
s.sharif@uel.ac.uk

***Abstract*— The use of chatbots has become increasingly popular in recent years, as more organisations try to improve and streamline their customer service operations. One area which has been gaining momentum is the use of chatbots for music recommendation. Such systems utilise AI technologies to deliver personalised music recommendations to users via conversational interfaces. Chatbot music recommender systems present several benefits namely; they provide a personalised and natural experience which can be engaging for the users. Moreover, the users can engage in a dialogue whereby the system can better interpret the user context and preferences. This work presents the development of a chatbot personalised music recommender system, based on Natural Language Processing (NLP) techniques, coupled with a web interface that can provide song recommendations based on the user's emotions.**

Keywords—chatbot music recommender system, Natural Language Processing (NLP), personalised music recommendation, NLP for music recommendation. Conversational music recommendation

I. INTRODUCTION

Music has been an integral part of human life since ancient times, and with technological advancements, our music consumption has transformed drastically. The availability of countless music options through platforms like music streaming services, social media, and online radios has revolutionized our listening habits.

In today's world, a song recommender system that considers users' emotions holds significant importance. Our mood greatly influences our music preferences, and a system that understands and aligns with our emotional state can create a more personalised and enriching music experience. Moreover, this system can have therapeutic effects by providing music that helps alleviate negative emotions, positively impacting mental health and well-being. The music industry's vastness has led to the rise of numerous music recommender systems, which suggest music to users based on their preferences and listening patterns. Popular platforms like Spotify [1], Pandora [2], and Apple Music [3] have further popularized these systems. The objective of such systems is to offer personalised and relevant music recommendations to users, introducing them to new music that resonates with their tastes. Ongoing research in this area aims to enhance recommendation accuracy, user experience, and inclusivity.

This work focuses on the development a chatbot song recommender system that delivers personalised music suggestions based on users' emotions. Machine learning algorithms, including collaborative filtering and content-based filtering, will be employed to generate recommendations. The proposed system has been evaluated based on accuracy, diversity, and user satisfaction using a dataset of music ratings and listening histories. The results shed light on the system's effectiveness and its potential to improve music discovery and recommendation.

Ultimately, this research aims to contribute to the field of music recommender systems by creating an effective, personalised song recommender system that caters to user preferences. By exploring the application of machine learning techniques, this work opens doors for further advancements in music recommendation research.

The remainder of the paper is organised as follows; Section 2 presents the state of the art in music recommender systems, Section 3 illustrates the system architecture, Section 4 describes the development of the model and the results followed by the Conclusion in Sections 5.

II. BACKGROUND

The rise of digital music consumption in recent years has led to the increasing popularity of song recommender systems. These systems aim to provide personalized music recommendations to users based on their emotional preferences and listening history. While the idea of personalised recommendations dates to the 1990s [4], it wasn't until the first decade of the twenty-first century that the first music recommendation algorithms were developed. One of the pioneering projects in this domain was the Music Genome Project, introduced by Pandora in 2000 [5]. This project analyzed song qualities such as melody, harmony, and lyrics to create custom playlists for users. Since then, various playlist-based music recommendation services, such as Spotify's Discover Weekly, Apple Music's Recommendations, and Amazon Music's Recommended Playlists [6], have emerged, leveraging user data and machine learning algorithms to provide tailored music suggestions.

The Music Genome Project revolutionized the music recommendation landscape by diving deep into the essence of songs. By dissecting and analyzing song characteristics, Pandora's recommendation engine identified patterns that appealed to users with specific preferences. These qualities included elements like genre, instrumentation, tempo, mood, and more. The system then used this knowledge to suggest songs with similar attributes to users who enjoyed tracks, creating personalized playlists.

Building upon the principles set forth by the Music Genome Project, modern music streaming services like Spotify, Apple Music, and Amazon Music have taken personalised recommendations to new heights. These platforms leverage vast amounts of user data, including listening history, song ratings, skip behavior, and even user-created playlists, to understand individual preferences better. This data is combined with sophisticated machine learning algorithms that can process and analyze these vast datasets efficiently.

A. Music Recommendation Systems

Music Recommendation Systems comprise algorithms which offer users individualised music selections [7]. Such systems can be segmented into three categories namely; content-based, collaborative filtering and hybrid systems.

Content-based filtering [8] is an approach that involves recommending songs with similar characteristics to those that a user has already expressed an interest in. Systems which employ this technique take into account different attributes of music tracks such as tempo, genre and instrumentation. Moreover, these systems can provide recommendations for genres or songs that may not have a lot of data. Furthermore, they can provide personalised suggestions based on users' listening history and preferences. However, users who have variety of musical tastes may experience difficulties in using these systems.

Collaborative filtering [9] is a widely used technique which finds patterns by analysing the behavior of similar users. If two users share similar music preferences, the system will recommend songs that one user has enjoyed to the other user and vice versa. Collaborative filtering systems can provide a range of recommendations across different genres and styles. In addition, they can provide recommendations on new and popular songs based on the collective preferences of users. However, these systems may not be as effective in providing recommendations for niche genres.

Hybrid approaches [10] that combine collaborative and content-based filtering have also emerged, providing even more accurate and diverse recommendations.

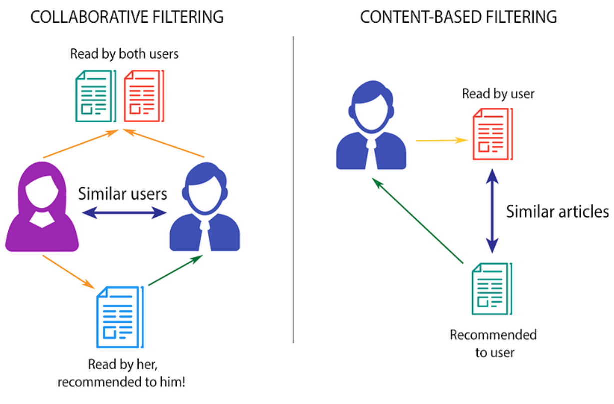


Fig. 1. Techniques used by recommender systems [11]

B. Natural Language Processing Techniques (NLP) for Music Recommendation

NLP techniques can be utilised for extracting and analysing information from textual data, such as song lyrics, user reviews, and social media posts. Some of the commonly used NLP methods for music recommendation are depicted in this section.

1) Sentiment Analysis

Sentiment analysis relates to the extraction of information about opinions, sentiments and emotions related to topics of interest [12]. Using sentiment analysis, the emotional tone of a song can be determined and as such, this information can be used to recommend songs of similar tones to users.

2) Topic Modelling

Topic modelling entails the use of unsupervised learning techniques for extracting main topics that appear within a collection of items. As such, this technique can be applied to recommend songs having a similar theme.

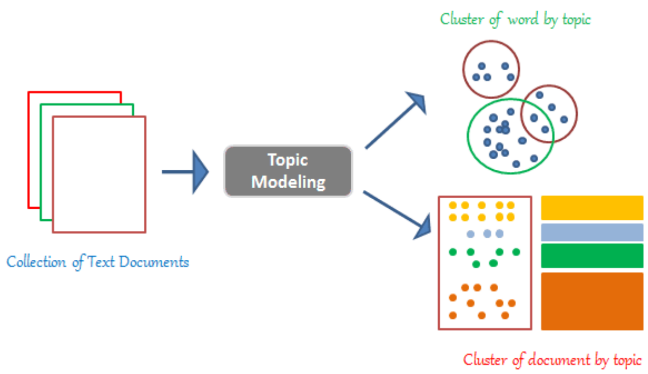


Fig. 2. Topic modelling [13]

III. System Architecture

This section presents the core components of the proposed system.

A. Chatbot Frontend

The front end plays a crucial role in creating an interactive and engaging user experience. The front end of a chatbot typically involves designing the graphical user interface (GUI) that users interact with, and HTML (Hypertext Markup Language) is a commonly used markup language for creating web-based chatbot interfaces.

B. Backend Server

Emotion detection, also known as sentiment analysis, is vital for businesses and organizations to understand customer feelings in text data. The backend server of the proposed system comprises technologies namely Sapling API [14], Last.fm API [15] and CakeChat Server [16]. The architecture of the proposed system is as shown in Fig. 3.

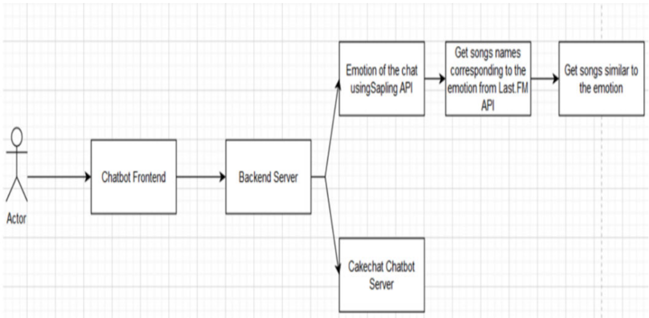


Fig. 3. Architecture of the proposed system

Sapling API offers powerful emotion detection capabilities for developers to integrate into their applications. It excels in various features, including NLP-based emotion identification, support for multiple languages, contextual understanding, customization, and easy integration. Utilising SAPLING API provides enhanced customer insights, real-time monitoring, personalization, and significant time and cost savings. Its versatile use cases span social media monitoring, customer service, market research, and brand management, making it an indispensable tool for analyzing emotions and sentiments in text data across industries and applications.

Cake Chat is an advanced conversational AI model developed by OpenAI. This neural network-based chatbot is trained on vast text data to emulate human-like interactions. It seamlessly engages with users across various platforms like web browsers, messaging apps, and APIs through the CakeChat chatbot server. Its remarkable features include handling multi-turn conversations with context retention for natural and dynamic chats. The chatbot's responses are customizable for specific use cases and can be refined through user feedback for continuous improvement. With API integration, developers can harness its potential for content creation, virtual assistants, customer service, and language learning. While generally reliable, some occasional limitations can be mitigated through fine-tuning and customization.

Last.fm, an online community and music streaming service, offers personalized song suggestions to customers based on their preferences. It provides a potent API for developers to access its data and services. Key features include retrieving artist information, track details, user data, search functionality, scrobbling, authentication, and rate limiting.

The Last.fm API finds use in various applications namely:

- Music Recommendation Apps: Customized suggestions and playlists based on listening habits and favorite artists.
- Music Players and Streaming Apps: Enhancing the music-listening experience with track and artist information.
- Social Music Apps: Allowing users to explore track details and artist information within the app.
- Analytics and Insights: Utilizing Last.fm API for gathering insights into user listening habits and music trends.

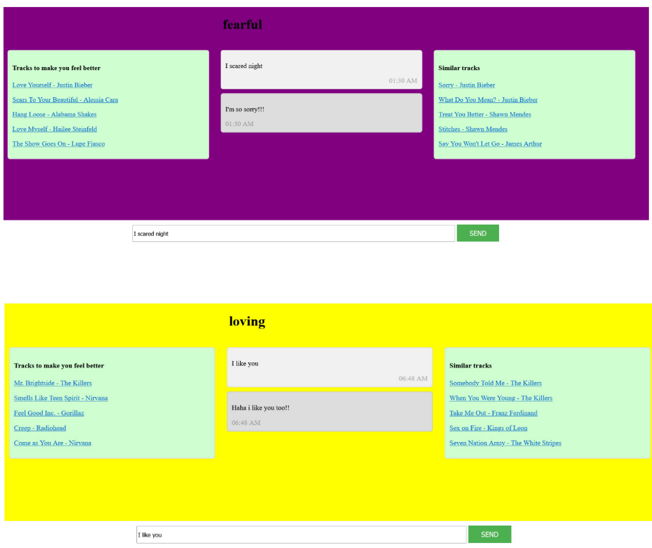
IV. System Implementation And Results

This Section depicts implementation of the proposed system as well as the interactions between the various core components.

The workflow initiates through the creation of a key for accessing the Sapling API. This key is then used to request data from a tone analyser which in turn, provided a tone response. The response is then validated, whereby the overall emotion is extracted and mapped to a corresponding emotion for the chatbot. This is followed by the generation of the chatbot request which is sent to the chatbot server as a HTTP request. A TonesMapper class is then used for emotion mapping whereby emotion characteristics such as happiness, sadness, rage, and fear are determined as well as the emotion intensity.

Following the emotion mapping, the music tracks corresponding to the emotions are then retrieved along with a colour which matches the emotion. Moreover, additional music tracks, which are similar to the recommended tracks, are returned.

Fig. 4 illustrate the user interface of the system whilst in operation. It provides a chat interface to allow the users to interact with the system and provide details on how they are currently feeling. The colour scheme is then adjusted based on the emotion detected. Furthermore, the recommended tracks pertaining to the user emotion are represented on the left side of the interface and additional similar tracks are listed on the right hand side.



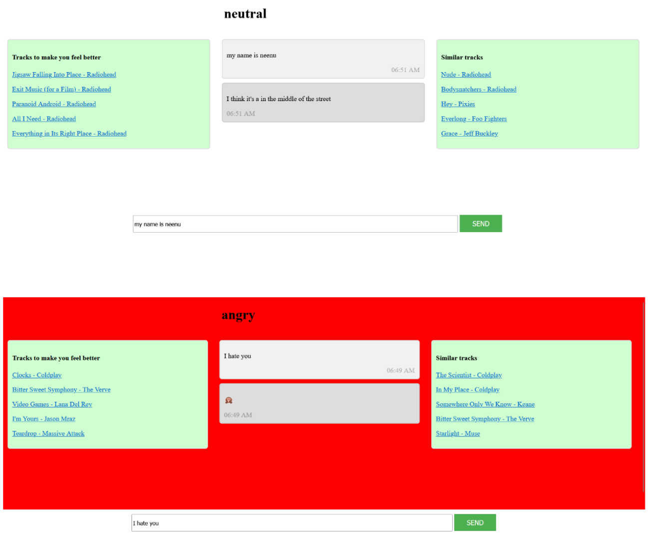


Fig. 4. User interface colour adaption and music track recommendation based on user emotion

V. CONCLUSION AND FUTURE WORK

This work presented the development of a chatbot music recommendation system using NLP techniques. The system features a web based front end and a backend server comprising of Sapling API, Last.fm API and a CakeChat server. The system allows users to enter messages related to how they are feeling and it outputs music tracks based on the users' emotions. The system has been tested for different types of emotions and is seen to provide appropriate music track recommends. As such, through personalised messaging and emotional recognition, the proposed system has potential for improving user satisfaction and loyalty.

Future work in this study entails integration with social media platforms to gather more data about users' emotions and preferences, allowing for a more personalised recommendation experience. Social media activity can be used to gain insights into users' wider interests and personalities, which can lead to more nuanced and accurate recommendations. By analysing users' interactions with music-related content on social platforms, recommender systems can understand their engagement levels and identify music trends more effectively.

In addition, contextual data, such as location, time of day, and weather conditions, can also be investigated as valuable input for music recommendations. For instance, a user may prefer upbeat, energetic songs during a morning workout but more relaxed tunes for winding down before bedtime. Incorporating contextual data can significantly improve the relevance and timeliness of suggestions, making the overall music streaming experience more enjoyable and immersive.

Further work will also include the development of a mobile application to provide users with recommendations via smart devices. Moreover, the dataset will be expanded to include more diverse genres and sub-genres of music, in order to make the system more comprehensive and cater to a wider range of users.

Looking ahead, the future of music recommender systems seems promising. As machine learning techniques continue to evolve, we can expect even more sophisticated algorithms capable of understanding complex user preferences and providing seamless, tailored music experiences. Additionally, advancements in natural language processing could enable voice-based music recommendations, allowing users to interact with the system in a more intuitive and conversational manner.

Ultimately, music recommender systems will continue to play a vital role in shaping the way we discover and enjoy music, connecting artists and audiences in new and exciting ways. The ongoing research and development in the field of artificial intelligence and machine learning will undoubtedly contribute to the continued success and evolution of music recommendation technology.

REFERENCES

[1] Spotify, "Spotify," *Spotify*. https://open.spotify.com/ (accessed Jul. 24, 2023).

[2] "Music and Podcasts, Free and On-Demand | Pandora," *Pandora*, 2000. https://www.pandora.com/ (accessed Jul. 24, 2023).

[3] Apple, "Apple Music," *Apple (United Kingdom)*, 2023. https://www.apple.com/uk/apple-music/ (accessed Jul. 24, 2023).

[4] P. Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56–58, Mar. 1997, doi: https://doi.org/10.1145/245108.245121.

[5] "The Inner Workings of the Music Genome Project : Networks Course blog for INFO 2040/CS 2850/Econ 2040/SOC 2090." https://blogs.cornell.edu/info2040/2012/09/23/the-inner-workings-of-the-music-genome-project/

[6] "Amazon Music Prime | Amazon.co.uk," *www.amazon.co.uk*. https://www.amazon.co.uk/music/prime/?_encoding=UTF8&ref_=sv_dmusic_2 (accessed Jul. 24, 2023).

[7] F. Ricci, L. Rokach, and B. Shapira, "Recommender Systems: Introduction and Challenges," *Recommender Systems Handbook*, pp. 1–34, 2015, doi: https://doi.org/10.1007/978-1-4899-7637-6_1.

[8] Ò. Celma, *Music Recommendation and Discovery*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. doi: https://doi.org/10.1007/978-3-642-13287-2.

[9] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, Dec. 1992, doi: https://doi.org/10.1145/138859.138867.

[10] Y. Qin and I. Fujinaga (Internal/Supervisor), "A historical survey of music recommendation systems: Towards evaluation," *escholarship.mcgill.ca*. 2013.

https://escholarship.mcgill.ca/concern/theses/7s75dh037 (accessed Jul. 24, 2023).

[11] S. Doshi, "Brief on Recommender Systems," *Medium*, Feb. 20, 2019. https://towardsdatascience.com/brief-on-recommender-systems-b86a1068a4dd (accessed Jul. 20, 2023).

[12] A. Yadollahi, A. G. Shahraki, and O. R. Zaiane, "Current State of Text Sentiment Analysis from Opinion to Emotion Mining," *ACM*

Computing Surveys, vol. 50, no. 2, pp. 1–33, May 2017, doi: https://doi.org/10.1145/3057270.

[13] "Topic Modelling With LDA -A Hands-on Introduction," *Analytics Vidhya*, Jul. 08, 2021. https://www.analyticsvidhya.com/blog/2021/07/topic-modelling-with-lda-a-hands-on-introduction/

[14] "Welcome | Sapling.ai Developer Documentation," *sapling.ai*. https://sapling.ai/docs/ (accessed Jul. 24, 2023).

[15] "API Docs," *Last.fm*. https://www.last.fm/api (accessed Jul. 23, 2023).

[16] "lukalabs/cakechat," *GitHub*, Nov. 08, 2022. https://github.com/lukalabs/cakechat (accessed Jul. 23, 2023).