International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 05 | May - 2022 Impact Factor: 7.185 ISSN: 2582-3930

SONG RECOMMENDER SYSTEM VIA CHATBOT

Dr. G. S. Bapiraju[1*], *A. Archana*[2], *A. Rasagna*[3], *B. Abhinaya*[4], *Roshini Guptha*[5]

[1,2,3,4,5] *Computer Science and Engineering*, *GRIET*, Hyderabad, Telangana, India.

Abstract. Music recommendation based on mood is much needed in this era of technological breakthroughs since it would assist humans to alleviate tension and listen to relaxing music based on their mood. We frequently spend our time on the internet chatting with numerous chatterboxes that are particularly created for such purposes or merely for pleasure. The chatbots include incorporated data that allows them to recognize and respond to the user's inquiry. We'll be merging different services and open-source tools in this project to create a Chatbot that recommends songs based on the tone of the user's conversation with the chatbot. Chatbots are designed to assist and scale company teams in their interactions with customers. It may exist in any major messaging program, such as Facebook Messenger, Slack, Telegram, Text Messages, and so on. Chatbot applications improve customer experience by streamlining interactions between people and services. At the same time, they provide businesses with new options to boost customer engagement and operational efficiency by lowering conventional customer service costs. This project focuses on creating a bespoke chatbot, which will be the first step in learning how to create your own professional chatbots.

INTRODUCTION

In general, we spend a lot of time on the internet chatting with various chat boxes, especially those geared toward such activities or just for fun. By evaluating and describing the intent of the user request, the chatbot recommends the right businesses. Natural language processing is used to answer user questions in a college research chatbot project. It is advantageous to users since it allows them to ask inquiries in their own language and receive responses fast.

You must be tired of the weird chatbots out there in the world which are made for mainly business purposes? 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.

EXISTING APPROACH

The purpose of chatbots is to support and scale business teams in their relations with customers. It could live in any major chat application 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 focused on building a custom chatbot that will be your fundamental step on the learning curve of building your own professional chatbots.

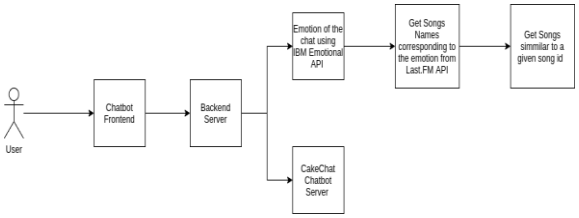
PROPOSED APPROACH

The proposed approach is an application that is primarily a Chatbot application that incorporates the emotion detection module. The emotion detection module is used for identifying the emotion expressed by the user and hence making it essential to the application as it provides entertainment in the form of Music according to the user's mood.

And here is the approach

User starts the conversation 1) Emotional Analysis of the conversation is done using the IBM Emotional API.2) Get the reply to the conversation from the Chatbot.3)Based on the Emotion which the app perceives, top songs are retrieved using Last.fm songs API .4)If a user listens to a particular song for some time, a similar song would be recommended to the user using Last.fm API.

SYSTEM ARCHITECTURE



A system architecture is a model which defines the structure, behavior, and perspectives of a system. An architecture interpretation is an authorized description and system representation, organized in such a way that supports reasoning about the structures and system behaviors. A system architecture consists of system components and the sub-systems developed, that will work together to implement the comprehensive system.

IMPLEMENTATION

Understanding the overall process involved in the chatbot working Once we are done with this milestone, we would be having a very good understanding of the complete project architecture. Further, after this, you can take this project in a new direction after this milestone. Completing this milestone is critical as there are a lot of components that need to be integrated together to complete the chatbot; having a clear sense of product architecture is needed here. • Have a look at the High- Level approach which we briefed earlier High-Level Approach - User starts the conversation - Emotional Analysis of the conversation is done - Get the reply to the conversation from the Chatbot
Based on the reply which we got, songs are retrieved We would be implementing this approach manually to get started right away by building the chatbot in the next milestone.

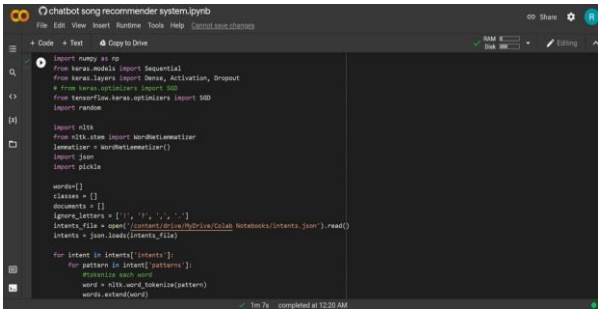
PROPOSED MODULES

The application consists of three main modules:

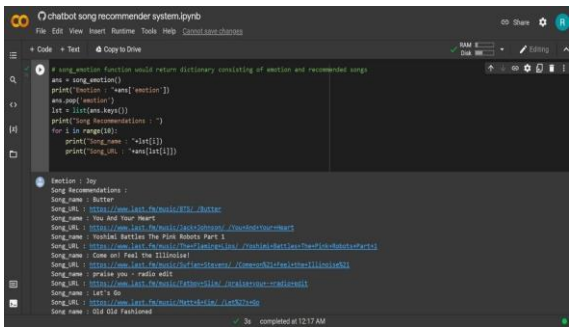
A. Chatbot. B. Emotion detection C. Music recommendation.

CHATBOT(Alex):

A chatbot is a software tool that uses text or text-to-speech to conduct an online chat conversation instead of providing direct contact with a real human agent. Alex is the name of our chatbot.



EMOTION DETECTION- IBM



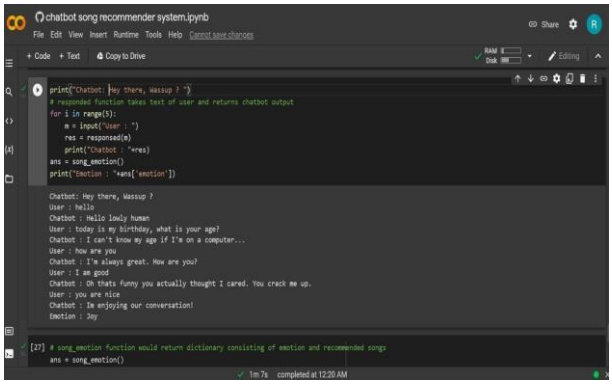
In this milestone, we would be setting up the IBM Tone Analyzer API so that we can analyze the tone of

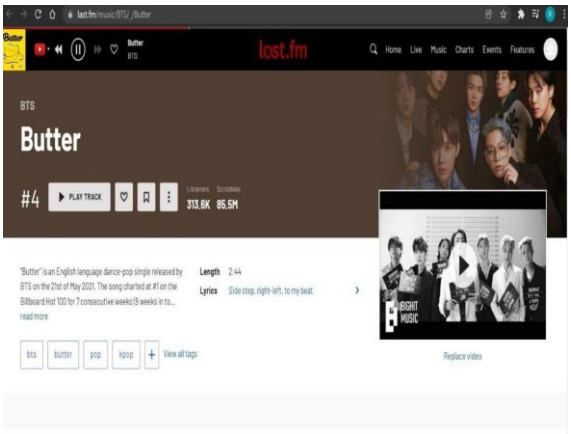
conservation(emotion). We are using an API here as we don't have enough data, Computational power, and time to create our own model API. This milestone will make you realize why to prefer using open-source API rather than creating your own models each time.

SETTING UP LAST.FM SONGS:

In these milestones, we would be setting up the Last.fm songs API so that we can recommend some songs to the user based on the emotion of the user. API is the acronym for application programming interface, which is a software intermediary that allows two applications to talk to each other. Each time you use an app like Facebook, send an instant message or check the weather on your phone, you are using an API. By the end of this milestone, you would have seen the response of both above-specified APIs Getting the top 5 tags of a particular tag.

RESULTS:





CONCLUSION

In our research we have proposed a system that automatically detects a person's mood or emotion and proposes music This suggested system employs a mapping method to assist the public in locating songs that match their emotional state. Our research also enables users to have a decent conversation with the system, which may ease some of their tension and may also propose songs. The proposed method can be deployed as an extra feature in many social networking platforms, adding the option of a chatbot. We devised a method for assessing persons that does not require any physical human intervention. This model, like the human brain, recognises human emotions and proposes songs to them.

Future Scope:

The main aim of the public is to stay stress-free. The main objective is entertainment for the people by ensuring that they are maintaining their stress levels or not.

- **Ensuring public stress balance:** The results obtained from the system can be used to suggest the songs for entertainment. This prevents people from worrying about their stress levels and gets entertainment from it.

- **Providing entertainment:** Basically, songs are for entertainment purposes. So by detecting the emotion or our present emotion if the songs are suggested, the user will be happier.

In the future, we would like to try the following things:

Using audio frequency to recommend songs.
Trying content-based algorithm
Making the recommender system a real-time system
Trying clustering techniques to recommend music.

REFERENCES

https://www.google.com/search?q=chatbot+song+recommender+system+project&sxsrf=AOaemvIIgXA0aQmFENray02v6IDk9JmGbA:1633025998404&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiyk6z0p6fzAhU36XMBHQ9UDAMQ_AUoA3oECAEQBQ&biw=1536&bih=722&d pr=1.25

https://www.ijert.org/chatbot-with-music-and-movie-recommendation-based-on-mood

https://towardsdatascience.com/create-music-recommendation-system-using-python- ce5401317159

https://www.google.com/search?q=chatbot+song+recommender+system+project HYPERLINK

https://www.google.com/search?q=chatbot+song+recommender+system+project&sxsrf=AOaemvIIgXA0aQmFENray02v6IDk9JmGbA:1633025998404&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiyk6z0p6fzAhU36XMBHQ9UDAMQ_AUoA3oECAEQBQ&biw=1536&bih=722&d pr=1.25"& HYPERLINK