**Deep Audiobook Tuner**

Submitted in partial fulfillment of the requirements

of the degree of

**B. E. Computer Engineering**

By

|  |  |  |
| --- | --- | --- |
| **Daniel Lobo** | **07** | **172028** |
| **Jenny Dcruz** | **16** | **172047** |
| **Smita Deulkar** | **19** | **172113** |
| **Leander Fernandes** | **20** | **172061** |

Guide:

**Ms. Priya Karunakaran**

Assistant Professor



Department of Computer Engineering

St. Francis Institute of Technology

(Engineering College)

University of Mumbai

2020-2021

**CERTIFICATE**

This is to certify that the project entitled **“Deep Audiobook Tuner”** is a bonafide work of **“Daniel Lobo” (Roll No. 07), “Jenny Dcruz” (Roll No. 16), “Smita Deulkar” (Roll No. 19), “Leander Fernandes” (Roll No. 20)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of B.E. in Computer Engineering.

**Ms. Priya Karunakaran**

**Guide**

**Dr. Kavita Sonawane Dr. Sincy George**

**Head of Department Principal**



****

**Project Report Approval for B.E.**

This project report entitled ***Deep Audiobook Tuner*** by ***Daniel Lobo, Jenny Dcruz, Smita Deulkar, Leander Fernandes*** is approved for the degree of ***B.E. in Computer Engineering.***

Examiners

1.---------------------------------------------

2.---------------------------------------------

Date:

Place:



Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed. 

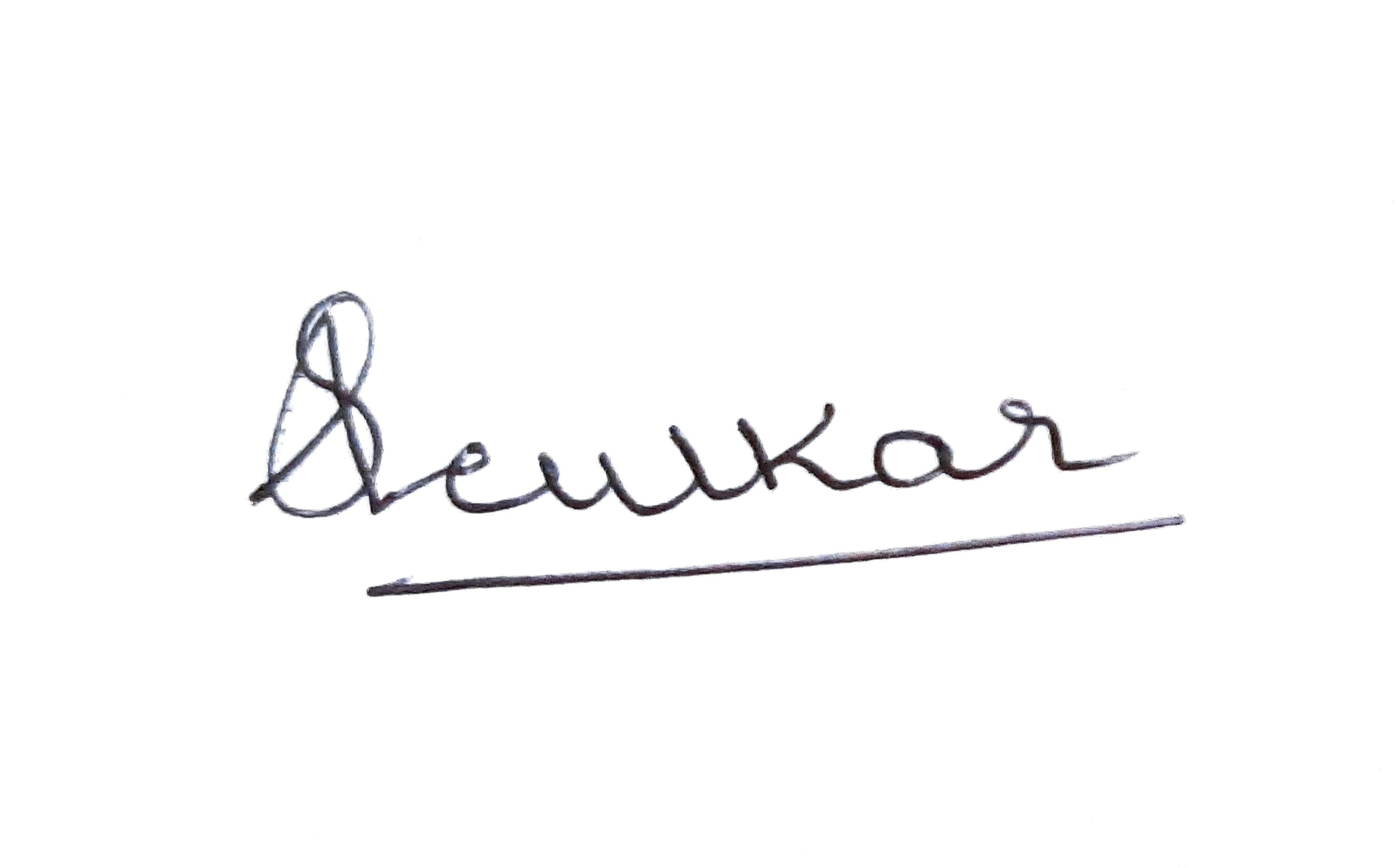
-----------------------------------------

Daniel Lobo (Roll No. 07)



-----------------------------------------

Jenny Dcruz (Roll No. 16)



-----------------------------------------

Smita Deulkar (Roll No. 19)



-----------------------------------------

Leander Fernandes (Roll No. 20)

Date:****

**Abstract**

Music plays a very important role in people’s lives. The music industry is worth $19 billion however the average person doesn’t have a vast knowledge of music theory. This makes it difficult for these people to come up with original and catchy melodies. Automating the creative process of the human means that companies can get multimedia products faster and cheaper. That is why music generators are such an important application of machine learning techniques. With the development of deep learning, neural networks are increasingly used in various art fields such as music, literature and pictures, and the artwork created can be even comparable to humans. Using a deep neural network, we can train a model to generate new melodies. Some of the different approaches that can be used to generate music using neural networks are LSTMs, bidirectional RNN, etc. Using Principal Component Analysis, we can adjust the component sliders to generate different melodies. This system has multiple applications. For example, musicians can use this application to generate new songs using the unique melodies that our system creates. In audiobooks, using sentiment analysis we can generate emotionally relevant background music. For our project, we will be focusing on creating an application that can take the text of a book as the input and generate relevant background music based on the predicted sentiment.

****

**Contents**

|  |  |  |  |
| --- | --- | --- | --- |
| **Chapter** | | **Contents** | **Page No.** |
| **1** |  | [**INTRODUCTION**](#_heading=h.autphoi5oyy1) | [**1**](#_heading=h.autphoi5oyy1) |
| **1.1** | [**Description**](#_heading=h.2g9eqth9buju) | [**1**](#_heading=h.2g9eqth9buju) |
| **1.2** | [**Problem Formulation**](#_heading=h.tyzlrhr3p3no) | [**2**](#_heading=h.tyzlrhr3p3no) |
| **1.3** | [**Objective**](#_heading=h.fyeko148qu1a) | [**3**](#_heading=h.fyeko148qu1a) |
| **1.4** | [**Proposed Solution**](#_heading=h.e2ek2q4mflhv) | [**3**](#_heading=h.e2ek2q4mflhv) |
| **1.5** | [**Scope**](#_heading=h.y184lac14xcm) | [**3**](#_heading=h.y184lac14xcm) |
| **2** |  | [**LITERATURE REVIEW**](#_heading=h.s6bcl82dpt40) | [**5**](#_heading=h.s6bcl82dpt40) |
| **3** |  | [**SYSTEM ANALYSIS**](#_heading=h.celhmfln9ypi) | **9** |
| **3.1** | [**Functional Requirements**](#_heading=h.6fa8dcw9lc92) | **9** |
| **3.2** | [**Non Functional Requirements**](#_heading=h.yrnbcilp52fo) | **9** |
| **3.3** | [**Specific Requirements**](#_heading=h.qoprl0mgossk) | **9** |
| **3.4** | [**Use-Case Diagrams and description**](#_heading=h.kkykzxlr6dlv) | **10** |
| **4** |  | [**ANALYSIS MODELING**](#_heading=h.b5u1rxuhya3q) | [**1**](#_heading=h.b5u1rxuhya3q)**3** |
| **4.1** | [**Class Diagram**](#_heading=h.lw3i7t81q4z3) | [**1**](#_heading=h.lw3i7t81q4z3)**3** |
| **4.2** | [**Functional Modeling**](#_heading=h.fus5kshkotpj) | [**1**](#_heading=h.fus5kshkotpj)**4** |
| **4.3** | [**Sequence Diagram**](#_heading=h.azj9emm243wx) | [**1**](#_heading=h.azj9emm243wx)**5** |
| **4.4** | [**Timeline Chart**](#_heading=h.dtkv1bjb96nv) | [**1**](#_heading=h.dtkv1bjb96nv)**6** |
| **5** |  | [**DESIGN**](#_heading=h.jy5ex4jvx5om) | [**1**](#_heading=h.jy5ex4jvx5om)**7** |
| **5.1** | [**Architectural Design**](#_heading=h.sqopo6c48eum) | [**1**](#_heading=h.sqopo6c48eum)**9** |
| **6** |  | [**CONCLUSION**](#_heading=h.wmuiztwv601q) | [**2**](#_heading=h.wmuiztwv601q)**1** |

References

Acknowledgements



**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Fig. No.** | **Figure Caption** | **Page No.** |
| 1.1 | Architectural Diagram | 2 |
| 3.1 | Use-Case Diagram | 10 |
| 4.1 | Class Diagram | 13 |
| 4.2 | Level 0 DFD | 14 |
| 4.3 | Level 1 DFD | 15 |
| 4.4 | Sequence Diagram | 16 |
| 4.5 | Timeline Chart | 16 |
| 5.1 | Architectural Diagram | 17 |
| 5.2 | Workflow of Client-end Software | 18 |
| 5.3 | Workflow for training Text-based sentiment analysis model | 19 |
| 5.4 | Workflow for training Audio-based sentiment analysis model | 20 |
| 5.5 | Workflow for Final sentiment analysis model | 20 |
| 5.6 | Workflow for training sentiment relevant Music Generation Model | 21 |

****

**List of Abbreviations**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Abbreviation** | **Expanded form** |
| i | TBSA | Text-Based Sentiment Analyser |
| ii | ABSA | Audio-Based Sentiment Analyser |
| iii | DFD | Data Flow Diagram |
| iv | VADER | Valence Aware Dictionary for Sentiment Reasoning |
| v | MFCC | Mel-frequency cepstral coefficients |
| vi | STFT | Short-time Fourier transform |
| vii | RAVDEES | The Ryerson Audio-Visual Database of Emotional Speech and Song |

****

**Chapter 1**

# Introduction

Oral storytelling is a thousand years old tradition and still abides till this day through digital recordings like podcasts, audiobooks, etc. An audiobook essentially is a recording of a book, novel, story, or other work being read aloud. They have been used extensively to teach critical listening, model good interpretive reading as well as aid students to understand books above their reading level and so on.

## Description

Audiobooks are being used on a regular basis by hundreds of users. The system in this report aims to develop emotionally relevant music for preexisting audiobook recordings. The user will enter an audiobook MP3 file as an input to the system. This audiobook will then go through two processes, simultaneously.

* First, the input audiobook will be run through a transcription tool to extract the text from the audiobook. This text will then be analysed using a Text-Based Sentiment Analyzer (TBSA).
* Concomitantly, in the second process, the features of the audio from the audiobook that is given by the user, will be extracted. The audio features are then analysed by an Audio-Based Sentiment Analyzer (ABSA) that will predict the emotions being conveyed in the audio.

Now the system will have obtained 2 values (sentiments) predicted by both, the TBSA as well as the ABSA. The values may vary and lead to an error. To avoid this, the weighted average of values will be calculated in order to generate the final predicted sentiments. Utilizing these predicted sentiments as well as the music generation model that has been explained ahead in this report, our application generates a seamless, distinctive musical score for every segment. These scores are stitched together along with the input audio file to provide the user an audiobook with felicitous background tunes.

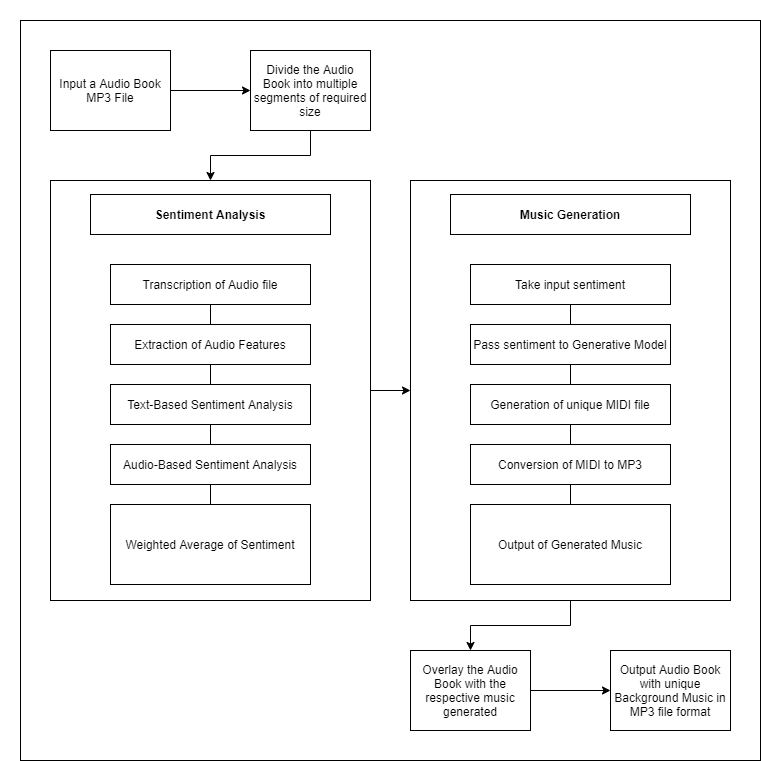


Fig. 1.1 Architectural Diagram

## Problem Formulation

The speech recordings that exist in the form of audiobooks are relevant but not as impactful. One may seldom find audiobooks with background music. Ordinarily, audiobooks consist of a monotonous speaker that reads the script to a book. Even though the content is being spoken out, the sentiments portrayed are not understood by the user. For example, a sarcastic line, if spoken without voice modulation may be perceived as offensive rather than humorous.

Therefore, adding a musical score to the audiobooks will enhance its effectiveness and make a difference in the user’s experience.

Crafting a musical score generator involves smoothly resequencing, looping, and timing the music to match the emotions in the story as they change over the course of the narrative. This is a challenging task even for experts. Due to this most audiobooks today only consist of speech. Existing audio editing tools force story producers to manipulate speech and music tracks using low level waveform editing which are expensive at times.

## Objective

To develop a system that generates an apt, emotionally pertinent, unique musical score for an audiobook automatically based on the current narrative for the purpose of ameliorating user-experience while being accurate, cost-efficient, and time saving.

## Proposed Solution

* Our system consists of two models, sentiment analysis model and a music generation model.
* The initial model will take an audio book file (script, audio) as it's input .
* It will segment the script.
* Individual segments are processed and topics are extracted from them.
* The sentiment for each segment is analyzed from the clusters of topics extracted.
* These sentiments are passed on as the input to the music generation model.
* In the second model, music is generated and the sentiments are an important factor in adjusting the components of our model.
* Sentiment is perceived in music due to several features such as melody, harmony, tempo, timbre, etc. Quieter volume, slower tempo and lower pitch is found to convey the emotion of sadness whereas louder volume, faster tempo and higher pitch is perceived as happier.

## Scope

This system aims to generate a unique background music score for audiobooks which is tailored towards the emotions being conveyed by the script at the moment. It is not feasible for every audiobook to have a unique background music score which is composed by a professional. The length of time a composer has to write the score varies from as little as two weeks or as much as three months to write the score. Professionals also charge per minute of finished music composition. Rates usually run from $50 to $1000 per minute of finished music. Hence this route is inefficient in time as well as cost. Our system aims to circumvent this process by using machine learning to generate a unique clip of music for every sentiment conveyed in the audiobook. These clips of music will then be overlaid on the respective segments of the audiobook which convey its sentiments to form the final soundtrack of the audiobook.

**Chapter 2**

# Literature Review

Review of literature is based on the previous work on Affective Algorithmic Music Composition, more specifically to works that process music in symbolic form in order to generate music with a given emotion. Music is a form of art conveyed through the medium of sound. General definitions of music include common elements such as pitch (which governs melody and harmony), rhythm (and its associated concepts tempo, meter, and articulation), dynamics (loudness and softness), and the sonic qualities of timbre and texture (which are sometimes termed the "color" of a musical sound). The emotion experienced in music is contributed highly by the structural features in the music. Structural features are divided into two parts, i.e. segmental features and suprasegmental features. Segmental features are the individual sounds or tones that make up the music, this includes acoustic structures such as duration, amplitude, and pitch. Suprasegmental features are the foundational structures of a piece, such as melody, tempo and rhythm. Tempo is typically regarded as one of the most important features affecting the emotion of the music, but a number of other factors, such as mode, loudness, and melody, also influence the perception of emotion in the piece.

Generating emotionally relevant music for videos and books is a problem that people have attempted to solve in different ways. A common approach for this problem consists of designing a rule-based system to map musical features to a given emotion in a categorical or dimensional space [1]. Steve Rubin and Maneesh Agrawala [2] proposed a system that would resequence music files in order to create a soundtrack that conveys the emotion perceived from an audio book. Their method requires premade music files for different emotions and worked best when the transcript of the audio book was labeled by humans.

## 2.1 Sentiment analysis of the audiobook

Sentiment analysis is a topic that has been worked on extensively over the past few decades. Most of the work in it focuses on textual data for analysis.

### 2.1.1 Sentiment analysis based on text

A sentiment analysis system designed for text based analysis combines two different techniques namely Natural Language Processing and Machine Learning. One such system is VADER ( Valence Aware Dictionary for Sentiment Reasoning). It is a sentiment lexicon model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion. A sentiment lexicon is a list of lexical features such as words which are generally labeled according to their semantic orientation as either positive or negative. VADER sentiment analysis relies on a dictionary that maps lexical features to emotion intensities known as sentiment scores. The sentiment score of a text can be obtained by summing up the intensity of each word in the text.

C.J. Hutto and Eric Gilbert [8] describe the development, validation, and evaluation of VADER in their paper. They used a combination of qualitative and quantitative methods to produce and validate a sentiment lexicon that was then combined with some generalizable rules regarding grammatical and syntactical conventions that humans use when expressing or emphasizing sentiment intensity. This combination of features and rules improved the accuracy of their sentiment analysis model across several different situations. VADER differentiates itself from other sentiment lexicons such as LIWC in that it is more sensitive to sentiment expressions in social media contexts while also generalizing more favorably to other domains. VADER can be used in cases of web sites wherein the text data could be a complex mixture of a range of text.

This method of analysis may not always accurately recognize the context of the situation described in the text (sarcasm, joy, sadness, etc). Sentiment analysis on audio introduces additional features such as tone, pitch, timbre. But audio sentiment analysis is still in a nascent stage in the research community. Using an hybrid approach comprising of audio analysis as well as text analysis would greatly improve the accuracy with which the system can extract emotions.

### 2.2.2 Sentiment analysis based on audio

Navas, Eva, Inma Hernez, and Iker Luengo [4] have shown that these prosodic features and acoustic features such as power and pitch contribute to the sentiment variation. By incorporating these audio features of the accompanying speech, the capability of the model to recognize the aforementioned context can be increased.

Arpit Shah and Shivani Firodiya [3] used acoustic features such as MFCC, STFT, Contrast, Mel Spectrum, Chroma and Tonnetz extracted from the audio clips of the RAVDESS dataset to train their fully connected DNN model for sentiment analysis on audio.

Classification of sentiments or emotions can be achieved through different measures. A popular approach is Russell's circumplex model [5] which classifies emotions based on their valence and arousal values. The valence dimension indicates whether an emotion is positive or negative, whereas arousal indicates the intensity of the emotion.

## 2.2 Generation of Music with sentiments

There have been multiple studies in the domain of music generation. However, generating music that conveys a certain sentiment has been a major challenge. The goal of Affective Music Composition (AMC) is to automatically generate music that is perceived to have specific emotions. Sentiment is perceived in music due to several features such as melody, harmony, tempo, timbre, etc. Tempo (the speed or pace of a musical piece) is highly associated with the emotion perceived in music. Faster tempo is generally perceived as exciting, happy or angry, whereas slower tempo is perceived as sad or serene. Other features that contribute to the emotions conveyed in music are the scale or key of the music, the intensity or loudness of the music and the pitch of the music. The volume (loudness) of the music conveys the intensity of the emotion. The minor scale conveys negative emotions whereas the major scale conveys positive emotions. In general quieter volume, slower tempo and lower pitch is found to convey the emotion of sadness whereas louder volume, faster tempo and higher pitch is perceived as happier.

MIDI files aren't like regular audio files. They are smaller in size and don’t contain actual audio data. These files explain what notes are being played as they are played, along with the duration and intensity of each note [6]. MIDI files are instructions for how the music should be produced, when a program tries to interpret the file. Advantages of MIDI include small file size, ease of modification and manipulation and a wide choice of electronic instruments and synthesizer or digitally-sampled sounds. MIDI files can be opened by a variety of programs. Some of them are Windows Media Player, Winamp, VLC, ect. In order to convert a MIDI file into a mp3 file a free file converter program called FileZigZag can be used.

Ferreira and Whitehead [1] have presented a generative mLSTM model that can be controlled to generate symbolic music with a given sentiment. Their model had been trained on a corpus of MIDI files. They treated the music composition problem as a language modeling problem and hence represented music pieces as a sequence of words and punctuation marks from a vocabulary that represents events retrieved from the MIDI file. Their model used a Genetic Algorithm (GA) to optimize the weights of their neurons in order to lead the mLSTM to generate only positive or negative pieces. Two independent executions of this GA were performed, one to optimize the mLSTM for generating positive pieces of music and another was for negative pieces. This model could also be used to perform sentiment analysis of symbolic music.

In another paper, Jessie Salas [7] has presented a new software Tambr that translates text into sound with the help of multiple synthesized voices selected in a way that their timbre associates with the sense and sentiment of the scene portrayed in the plot. In order to get topics from a text, Tambr uses a human-labeled training data based search engine along with word-embeddings extracted from Google Word2Vec and implements a topic extraction algorithm that subsequently finds synthesizers that sound the most like the topics. Further, it engenders the sequence of pitches using Pantelis N. Vassilakis’ proposed system based on a linearity of dissonance. The dissonance or roughness of the intervals in the chords is dependent on the negativity of the sentiment at a certain point of the novel. Upon attaining the sentiment distribution, Tambr takes the statistical variance of the whole list. The dramatic plot shifts as well as the emotional density are represented by the high sentimental variance in the novels. This is accounted for by selecting the number of notes in a chord-voicing substantial for novels with supplemental emotionally diversive sentiments.

**Chapter 3**

# System Analysis

## 3.1 Functional requirements

* The system will allow the user to upload an audiobook MP3 file.
* The system will transcribe audio and extract features.
* The system will perform sentiment analysis and find predicted sentiments which will be used to generate music.
* Based on these predicted sentiments, emotionally relevant music will be generated.
* The system will overlay the input audiobook with the emotionally relevant background musical score.

## 3.2 Non-Functional requirements

* Performance: The response time of the system will be relatively low i.e., the text will be analyzed and music will be generated according to the emotions in a few seconds.
* Scalability: The system will work for small as well as large audiobooks as analysis is done according to segments.
* Reliability: The reliability of the system will be dependent on the accuracy of the sentiment analysis model as well as the music generation model.
* Maintainability: The system will be developed such that it can easily extend to incorporate additional functionalities.
* Availability: The system will be available to the user 24x7.
* Usability: The system will provide a user-friendly interface.

## 3.3 Specific requirements

**Software Requirements:**

Operating System: Windows 10

Python 3

**Hardware Requirements:**

CPU: Intel i5/Ryzen 5 or above

Clock speed: 2.5 GHz or above

GPU: NVIDIA Geforce GTX 960 or above

RAM size: 8GB or above

Hard Disk capacity: 100GB or above

## 3.4 Use-Case diagram and description

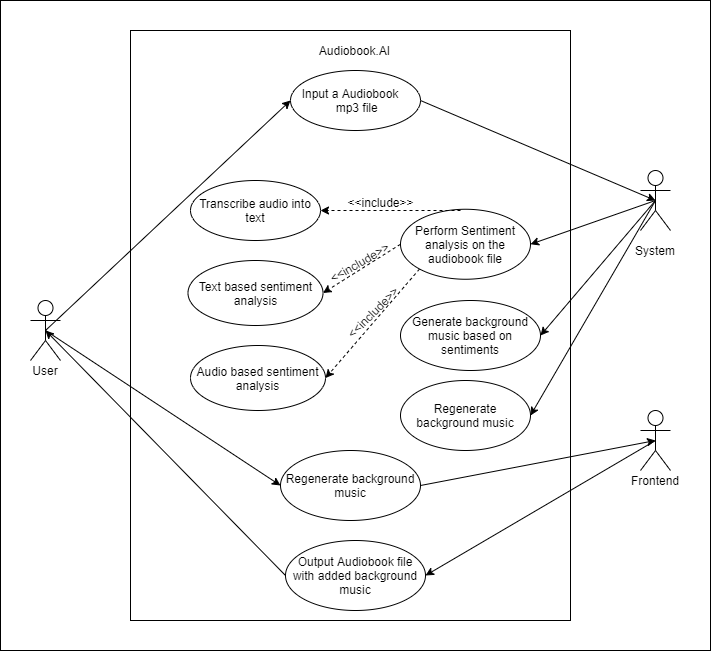
****

Fig. 3.1 Use-Case diagram

**Use Case Diagram:**

A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and the use cases. Use cases are a set of actions, services, and functions that the system needs to perform. In this context, a “System” is something being developed and operated, such as a website. The “Actors” are people or entities operating under defined roles within the system. An example of this with reference to this system is shown in Fig. 3.1.

**Use Case Specifications:**

* **Use case:** Input an audiobook mp3 file.

**Brief Description:** This use case will expect the user to upload an audiobook mp3 file as an input to the system.

**Primary Actor:** User

**Main Flow:**

* The frontend will provide an upload link for the user.
* The user has to upload an audiobook mp3 file via this link.
* **Use case:** Perform sentiment analysis on the audiobook file.

**Brief Description:** The system will perform sentiment analysis on the audiobook using a hybrid method which consists of both audio and text sentiment analysis.

**Primary Actor:** System

**Main Flow:**

* The system will segment the audiobook file into the specified time lengths.
* The system will then transcribe the audio segments to obtain the text from it.
* It will then perform audio sentiment analysis on the segmented audiobook mp3 files and on the segmented text files simultaneously.
* The system will then take the average of both these analyses and determine the conveyed emotion for each segment of the audiobook file.
* **Use case:** Generate background music based on sentiments.

**Brief Description:** The system will generate different short clips of music, each corresponding to a classified sentiment. The music generated will convey the same emotion which was used to generate it.

**Primary Actor:** System

**Main Flow:**

* The system will adjust the components of the deep neural network depending on the sentiment for which it has to generate music.
* Using the adjusted weights, the system will generate short music clips for all the classified sentiments.
* **Use case:** Output audiobook file with added background music.

**Brief Description:** The system will join the various clips of music with smooth transitions and create a background soundtrack.

**Primary Actor:** System

**Main Flow:**

* The music clips are joined to each other according to the emotion determined for each segment of the audiobook file.
* The transition between each clip will be smooth and a single music file will be created.
* The audio file is combined with the music file to create one output file which is the audiobook with background music.
* **Use case:** Regenerate background music.

**Brief Description:** If the user isn't satisfied with the background soundtrack then they can tell the system to regenerate the music.

**Primary Actor:** User

**Main Flow:**

* The user will click on the regenerate music button on the frontend if they are displeased with the soundtrack.
* The system will then regenerate music in the same manner.
* The new output file containing the regenerated music will be provided to the user.

**Chapter 4**

# Analysis Modeling

## 4.1 Class Diagram

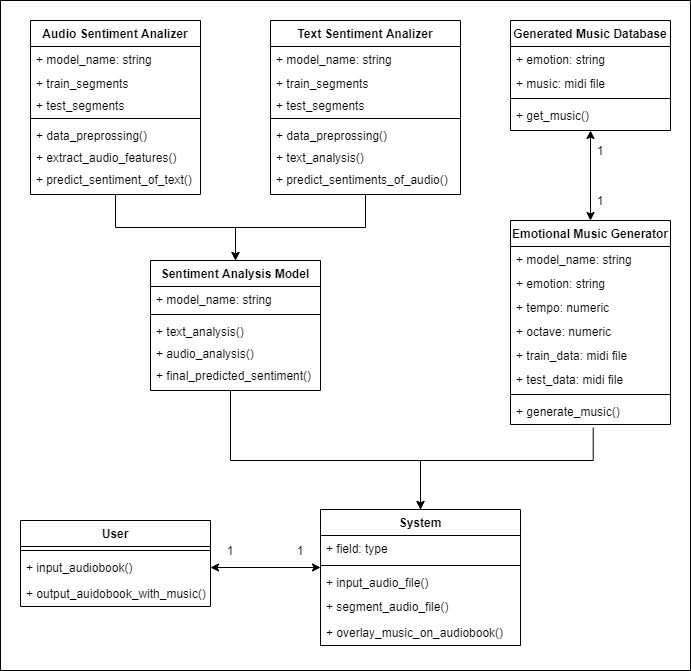
****

Fig. 4.1 Class diagram

Our project consists of the following classes as displayed in Fig. 4.1:

* **User**: The user can upload an audiobook file. Once it has been processed, the user can listen to the audiobook file with the added background music.
* **System:** The system can receive the audiobook mp3 file as its input. It will then segment the audiobook file and preprocess it. When the sentiments for each segment are predicted and the relevant music is generated, the system will join the music clips with smooth transitions and overlay the audiobook file with the background music.
* **Sentiment Analysis Model:** The Sentiment Analysis model is a hybrid model that makes use of an ABSA and a TBSA. Once it receives the sentiments from both these analysers, it will take a weighted average and decide the final sentiment.
* **Audio Based Sentiment Analyzer:** The ABSA will perform audio sentiment analysis on the segments of the audiobook. It will predict a sentiment for each segment.
* **Text Based Sentiment Analyzer:** The TBSA transcribes each segment of the audiobook file in order to obtain the text format. It will then perform text sentiment analysis on each text segment and hence determine a sentiment for each segment.
* **Emotional Music Generator:** The emotional music generator will generate a clip of emotional music for each sentiment predicted by the Sentiment Analysis Model.
* **Generated Music Database:** The music generated by the Emotional Music Generator will be stored in this database. The system will use the music clips in this database to overlay the audio file with the emotional background music.

## 4.2 Functional Modeling

Fig. 4.2 displays the level 0 Data Flow Diagram (DFD) of the Deep Audiobook Tuner system. The User gives a MP3 Audiobook file to the system as an input. The sentiments of the audiobook are then analysed. Using these sentiments, emotionally pertinent music is generated and overlaid with the audiobook.

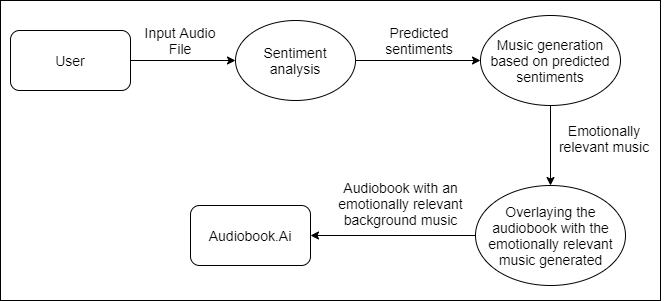
****

Fig. 4.2 Level 0 DFD

Fig. 4.3 shows the level 1 DFD of the Deep Audiobook Tuner system. It describes the system more elaborately. The input audiobook file goes through 2 processes. In one process the text is extracted from the audiobook, analysed and sentiments are predicted by the TBSA. Simultaneously in another process, the input audiobook file’s features are extracted and the sentiments are analysed by the ABSA. The average of these predicted values are considered for music generation. The music generated is then overlaid with the audiobook and presented as an output to the user.

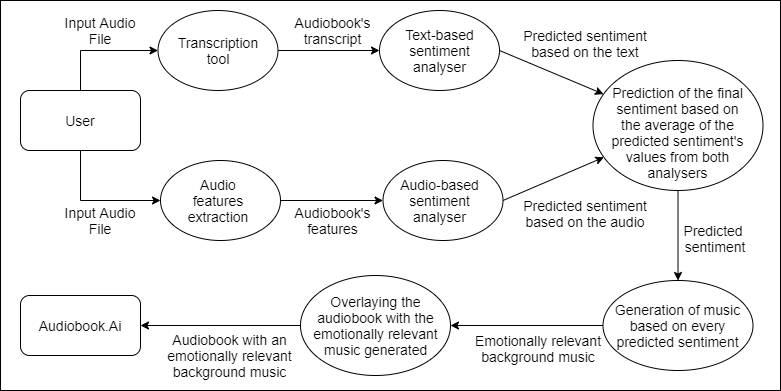


Fig. 4.3 Level 1 DFD

## 4.3 Sequence Diagram

Fig. 4.4 represents the sequence diagram of the Deep Audiobook Tuner application. The user inputs the audiobook MP3 file and this information is forwarded to the system which is used to transcribe audio as well as extract features from the audio. The transcribed text and extracted features are then passed through the sentiment analyzer to generate predicted sentiments. The music generator generates emotionally relevant music based on the predicted sentiments. The system overlays the audiobook with generated emotionally relevant music. The system then produces an audiobook with an emotionally relevant background musical score as output to the user.

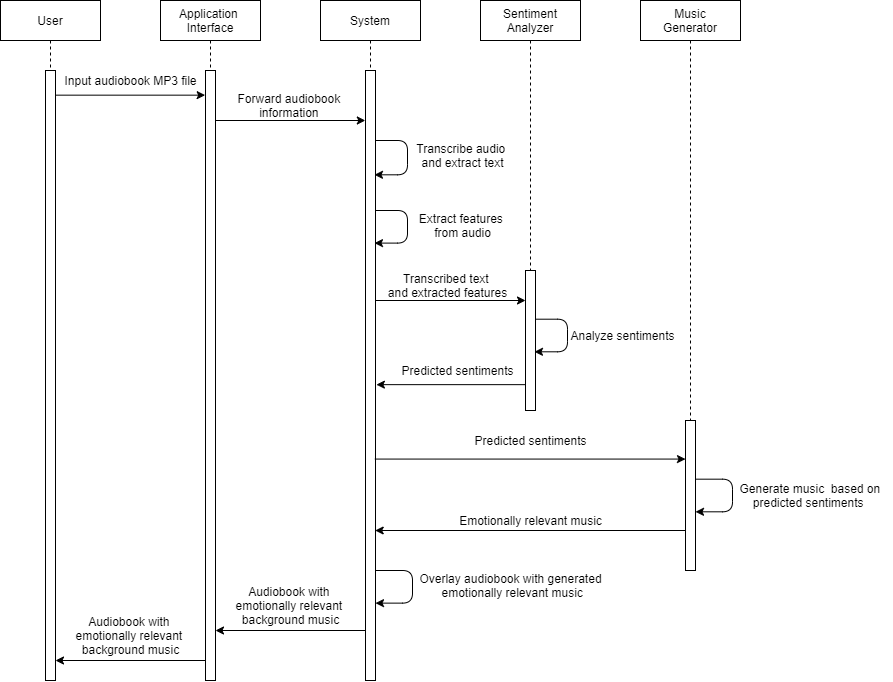
****

Fig. 4.4 Sequence Diagram

## 4.4 Timeline Chart

Fig. 4.5 displays a Timeline Chart that consists of the list of events in chronological order. It is a visualization of the tasks involved in the development of the Deep Audiobook Tuner system. The second column lists all the tasks. The adjoining column consists of long bars explaining the duration of the task with the respective dates in the first row of that column.

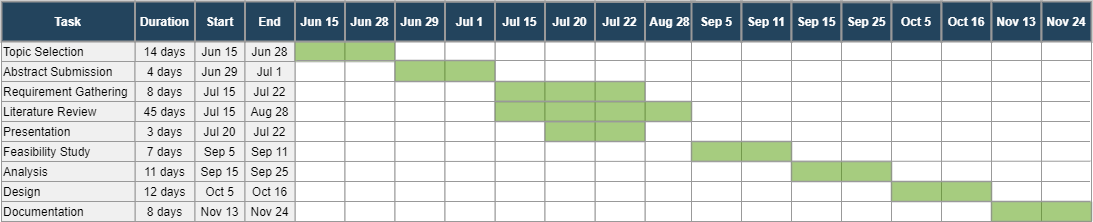
****

Fig. 4.5 Timeline Chart

**Chapter 5**

# Design

## 5.1 Architectural Design

Fig. 5.1 shows the architectural design of the application. It defines a structured solution to meet all the technical and operational requirements.

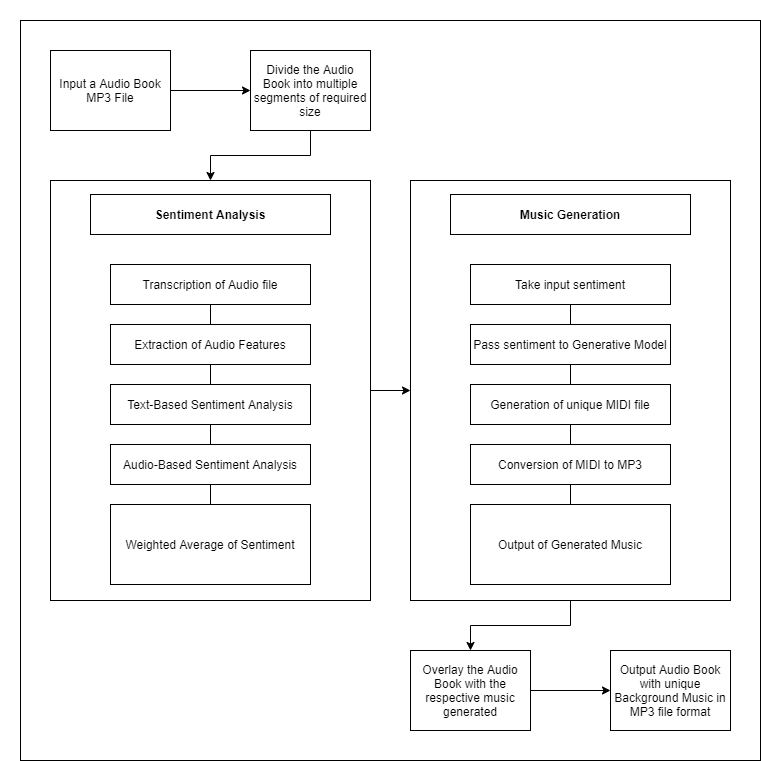


Fig. 5.1 Architectural Diagram

The user will select an audiobook for which he wishes to generate an emotionally relevant background music track. In the Sentiment Analysis phase this audiobook will then be run through a transcription tool to extract the text from the audio for the TBSA and audio features such as MFCC, zero crossing rate, etc. will be extracted for the ABSA. The weighted average of values will be calculated in order to generate the final predicted sentiments. In the Music Generation phase the sentiments detected by the Sentiment Analysis phase will be passed to the music generation model for it to generate unique MIDI files which are emotionally related to the given sentiment. The MIDI file will then be converted to a MP3 file for it to be overlaid with the audiobook. The final output of the system is an audio file with the audiobook overlaid with emotionally relevant background music.

### 5.1.1 Workflow of the Client-end Software

The overall workflow of the system is represented in Fig. 5.2, following which each model is explained in detail. The input provided by the user will be in the form of an MP3 file of an audiobook. This audiobook is then analysed by the system and divided into several segments of a fixed size. Following this, the sentiment portrayed in every segment will be predicted by the sentiment analysers. The system’s music generation model then generates music with respect to the sentiments predicted. These musical scores are then overlaid with the audiobook to provide the user an output MP3 file consisting of the audiobook in tandem with a musical background.

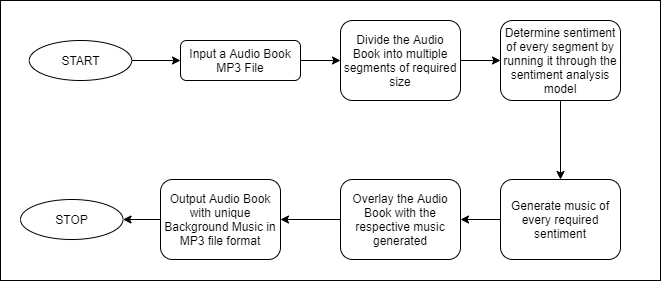


Fig. 5.2 Workflow of Client-end Software

### 5.1.2 Text-Based Sentiment Analysis (TBSA) model

The TBSA model is one of the two sentiment analysis models of the system represented in Fig. 5.3. This model runs alongside the ABSA model. The TBSA model is trained using a comprehensive dataset of several texts that consist of relevant labels of different emotions. For instance, a word like ‘Bravo’ will be labeled as happy, ‘unfortunately’ will be labeled as sad, ‘offended’ would be labeled as upset or angry and so on. This model shall be optimized by tweaking the hyperparameters in order to produce the most accurate output. The model with the least loss and highest accuracy will be chosen for the sentiment analysis of the text.

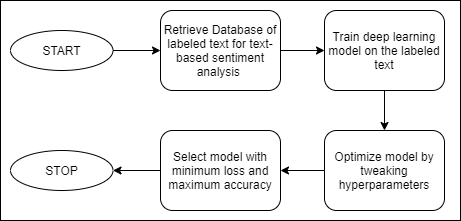


Fig. 5.3 Workflow for training Text-based sentiment analysis model

### 5.1.3 Audio-Based Sentiment Analysis (ABSA) model

The second sentiment analysis model is represented in Fig. 5.4, the ABSA is trained on a database of MP3 files that are labeled with respect to the emotions that the system takes into consideration. For instance, under the label ‘happy’ an upbeat MP3 audio file will exist. Similarly, for anger one will find the audio file with a high tempo and so forth. The audio features of these files will be extracted so as to train a deep learning model which essentially gives us our ABSA model. This model will be optimized like the previous TBSA model and the one with the least loss and most accuracy will be picked for sentiment analysis.

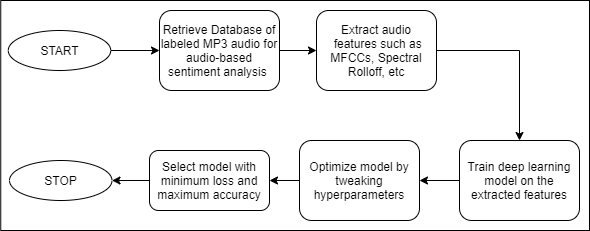


Fig. 5.4 Workflow for training Audio-based sentiment analysis model

### 

### 5.1.4 Final sentiment analysis model

The final sentiment model fundamentally uses the TBSA and ABSA models represented in Fig. 5.5. The sentiments predicted by using both these models may or may not be identical. To avoid any oversights, the system takes the weighted average of the predicted sentiments and produces a final sentiment with the most accuracy that is sent ahead to the music generation model.

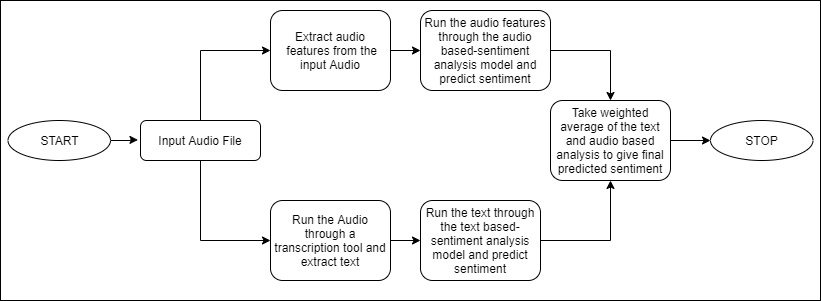


Fig. 5.5 Workflow for Final sentiment analysis model

### 5.1.5 Music generation model

Besides the sentiment analysis model, the system also consists of a music generation model represented in Fig. 5.6. In this, we train a GAN model on the extracted audio features of labeled Musical Instrument Digital Interface (MIDI) files that we retrieve from our database. Like we did with the other models, the music generation model is also then optimized by tweaking the hyperparameters. Upon completion of training, we select the model with the slightest loss and most accuracy. Using this model and our final predicted sentiments we generate several musical scores for every segment of the audiobook. These musical scores are stitched together and the music generated is overlaid with the input audiobook file to produce an output audiobook with unique background music that compliments the changes in the plot of the story to create an enhanced immersive experience for the listener.

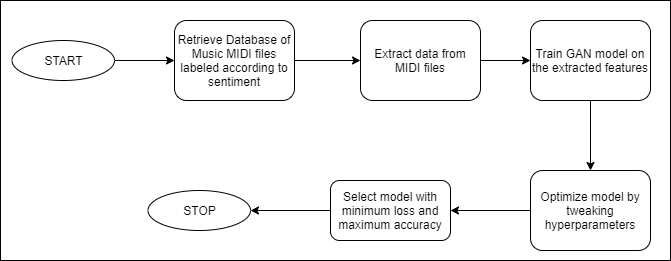


Fig. 5.6 Workflow for training sentiment relevant Music Generation Model

**Chapter 6**

# Conclusion

High-quality narrations are integral to creating captivating digital content, but most audio books do not have background musical scores. The few that do have it are expensive as it is a tedious task to create a soundtrack for it. Our proposed system can generate emotionally relevant scores which are cost effective, efficient and easily accessible to authors who want to add a soundtrack to their audiobook. This report has discussed the development of a background music generation system for the enhancement of audiobooks.

The objective of this system is to create a seamless system for enriching the users experience by performing sentiment analysis of audio and text for the generation of musical scores. Keeping this in mind we worked on transcribing the audio to text in order to perform text analysis. Transcription of the audio was facilitated by the Speech to Text AI provided by IBM Watson. This transcriber gives us a list of sentences for the specified segment length. Once we obtained the transcript to the audio, we ventured on to analyse our text. The text was analysed using VADER which relies on a dictionary that maps lexical features to emotion intensities known as sentiment scores. We obtained a sentiment score for each sentence and averaged it out for the entire segment. Thus the overall sentiment for each segment of the audio file was derived.

#### References

.

[1] Lucas N. Ferreira, Jim Whitehead. “Learning to Generate Music With Sentiment”, *20th International Society for Music Information Retrieval Conference, Delft, The Netherlands,* 2019.

[2] Steve Rubin and Maneesh Agrawala. 2014. “Generating emotionally relevant musical scores for audio stories”. In Proceedings of the 27th *annual ACM symposium on User interface software and technology (UIST '14).* Association for Computing Machinery, New York, NY, USA, 439–448.

[3] Arpit Shah, Shivani Firodiya, “Audio Sentiment Analysis after a Single-Channel Multiple Source Separation.” Indiana University Bloomington.

[4] Navas, Eva, Inma Hernez, and Iker Luengo. “An objective and subjective study of the role of semantics and prosodic features in building corpora for emotional TTS.” IEEE transactions on audio, speech, and language processing 14.4 (2006).

[5] Russell, J. A. A circumplex model of affect. Journal of personality and social psychology 39, 6 (1980), 1161.

[6] Tim Fisher, 2020, *What Is a MIDI File?*, Lifewire, accessed 30 October 2020, <<https://www.lifewire.com/midi-file-2621979>>

[7] J. Salas, "Generating Music from Literature Using Topic Extraction and Sentiment Analysis," in *IEEE Potentials*, vol. 37, no. 1, pp. 15-18, Jan.-Feb. 2018.

[8] C.J. Huttom, Eric Gilbert, “VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text”, Association for the Advancement of Artificial Intelligence (www.aaai.org). 2014.

#### Acknowledgements

A project is an activity that needs involvement of a lot of people from different cadres and calibres. Hence it would be highly unfair on our part to claim credit for the whole project. With due respect, we express our sincere gratitude to our esteemed Principal Dr. Sincy George for this teaming experience and for giving us an opportunity to work on this project.

We sincerely want to thank our project guide Ms. Priya Karunakaran for her valuable guidance and advice she has given us when faced difficulties while working on the project. We are also grateful for her constructive criticism and for helping us develop this idea to pursue the project. We would also like to thank our Class Coordinator Ms. Safa Hamdare for her notable inputs and the advice she has given us in our projects.

We would also like to thank our Head of the Department, Dr. Kavita Sonawane and the Computer Department staff who have helped us throughout this project. We are fortunate enough to get encouragement and support from all teaching staff of the computer engineering department. We would also like to extend our sincere regards to all the non-teaching staff of the computer department for their timely support.