IMPROVED BACKGROUND MUSIC FITMENT IN VIDEO POWERED BY MACHINE LEARNING

Punyabrota Dasgupta

AWS India

ABSTRACT

Background music scores are an integral part of movies or shows that we enjoy. It amplifies the impact and aesthetic appeal of the dialogues and scenes. With exponential growth in content production, what can be an agile, scalable and cost-effective mechanism to attach the best-fit background score for a scene? Relying and churning a musician's creativity every time is costly and slow.

Social media platforms pre-acquire music rights, and provide an asset library to content creators to choose from. What if the same can be extended for TV show production? For an intense visual scene, what will be the best background music? Beethoven's 5th symphony or an Indian classical raga clip? Can we match this by emotion determination, using ML?

In this solution, we analyze the dominant emotion of a video scene, through artists facial expressions, lighting conditions and dialogues. The dialogues are analyzed for transcript text, tonality such as pitch, loudness, pause, mid-level features such as spectrogram, MFCC, chroma, etc. The emotions are classified leveraging 'The Circumplex model of Emotions', in a 2D space of valence and arousal. Similarly, pre-acquired music tracks are analysed for high, mid and low level features to infer the best possible emotion is depicts. Classical music (example: Indian Ragas) do have well documented literatures, outlining the principal mood they emit. Such choices are deterministic and will have higher accuracy in mood labelling. Once the music track is chosen based on commonality of emotions, it may be possible to programmatically alter its tempo and pitch, to blend aesthetically into the video scene. Even if a musician has to play the same tune on a different instrument (different timbre), pitch, and tempo – it still does significantly reduce the production effort, time and cost.

.

INTRODUCTION

Background music in video content is an essential component to boost user experience and engagement. With the prolific growth in rate of content production, the cost of associated background music production also increases. Time and effort have to be budgeted separately. In this paper, we explore how machine learning can be used to extract the dominant emotion of a video segment and generate an appropriate background music for the same.

EMOTION CLASSIFICATION TAXONOMY

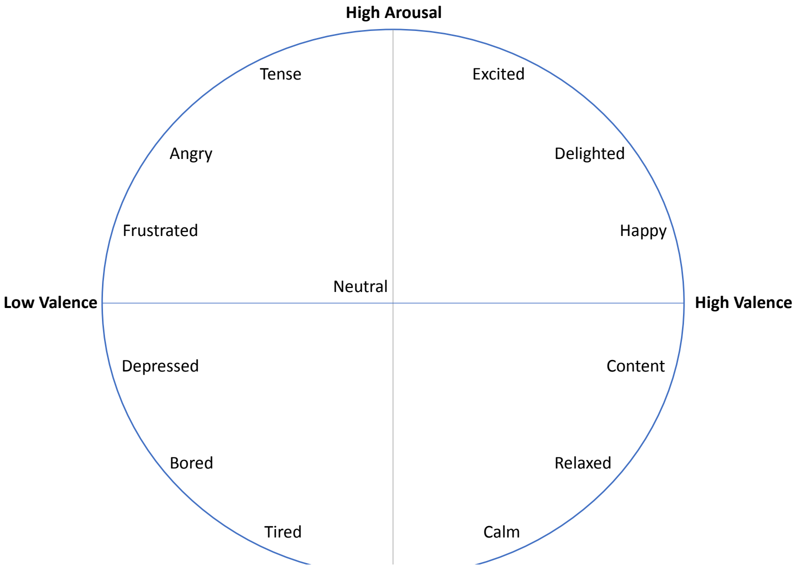


Figure 1 - Emotion Representation using Circumplex Model

A common emotion classification taxonomy is required to connect the emotions emitted from video scenes to the best fit emotion for the background music. The Circumplex Model of emotion classification is adopted as a foundational framework for this. As per this model, all emotions can be captured in a two-dimensional circular space, with the vertical axis representing arousal and the horizontal axis representing valence. Emotional states can be represented at any level of valence and arousal, or at a neutral level of one or both of these factors. Circumplex models have been most popularly used to test stimuli of emotion words and facial expressions. For example, a scene depicting a person receiving a surprise gift from a friend would mostly emit high arousal and high valence emotion, falling somewhere in the happiness band. Similarly, a scene depicting a lonely person idling throughout the day, with nothing to do would qualify in the low arousal, low valence space of being sad or bored. The adjacent diagram further illustrates this concept.

APPROACH TO EMOTION FEATURE EXTRACTION

A video segment is composed of visuals, audio and transcript. Each of these components may contribute to the overall emotional portrayal. The diagram below outlines how these video components are treated in parallel for extraction of features related to emotions. Let us dive deeper into how each such feature extraction is handled.

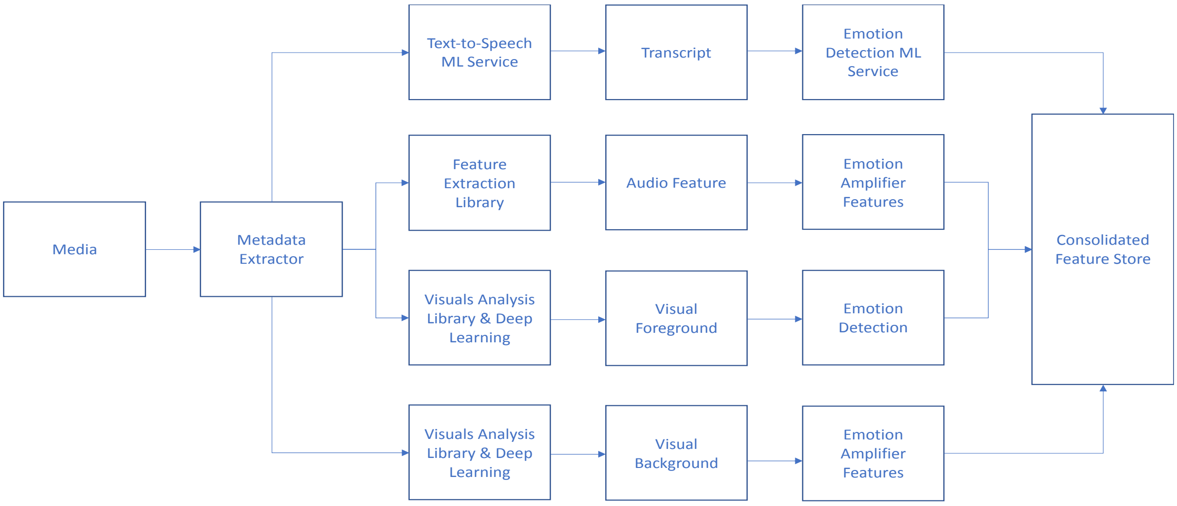

Text-to-Speech ML Service
Transcript
Emotion Detection ML Service
Feature Extraction Library
Audio Feature
Emotion Amplifier Features
Media
Metadata Extractor
Consolidated Feature Store
Visuals Analysis Library & Deep Learning
Visual Foreground
Emotion Detection
Visuals Analysis Library & Deep Learning
Visual Background
Emotion Amplifier Features

Figure 2 - Metadata extraction from video segment

Visual Analysis – The visual analysis is further divided into two planes. Human actors present on the frame are considered as the source for dominant visual emotion. the background is analyzed as auxiliary visual emotions. Amazon Rekognition is used for the visual analysis. The key analyses that are being done are 1/ face detection and associated emotional expression. 2/ foreground and background color in the frame, brightness, contrast, etc. For example, a dark background scene may have been used to paint a negative emotion. 3/ Label or object detection is done to gain insights of whether the scene is set to outdoor or indoors, or a place of tourist interest. 4/ The video segment is analyzed for rate of scene, segment, or shot change detection. For example, a high rate of shot changes may be reflective of tense or action sequence.

Depending on the project criticality and funding available, human labelers can be engaged through a crowd-sourcing model, to objectively label the dominant emotion of the video. For the purpose of this pilot, all such emotional attributes are collected, aggregated and normalized on a flat linear scale. In diverse production applications, some emotional features may be need to be amplified by adding suitable multiplier weights to them.

Audio Analysis – The conversations between different characters on the video segment does have two aspects. One is the tonality of the conversation and the other is the transcript or the content of the conversation. Let us explore the tonal aspects in more depth. The audio is analyzed using open-source python package Librosa. The two modes of analysis are on the time and frequency domain. On the time domain, rate of variations in the loudness and intensity can be reflective of an agitative dialogue. Analysis on the frequency domain is indicative of the voice pitch range, detection of being a male or female, conversation in a low pitch voice, a whisper, sobbing, laughter, etc. can be classified. The adjacent diagram illustrates the typical analysis of audio signal from a time and frequency domain perspective.

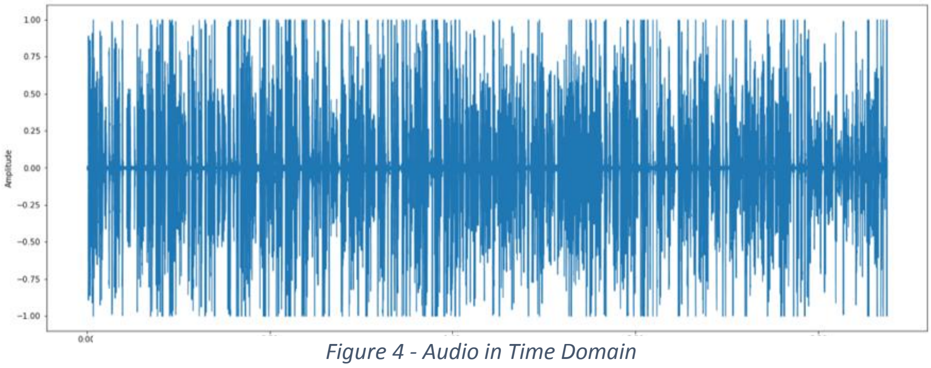


Figure 4 - Audio in Time Domain

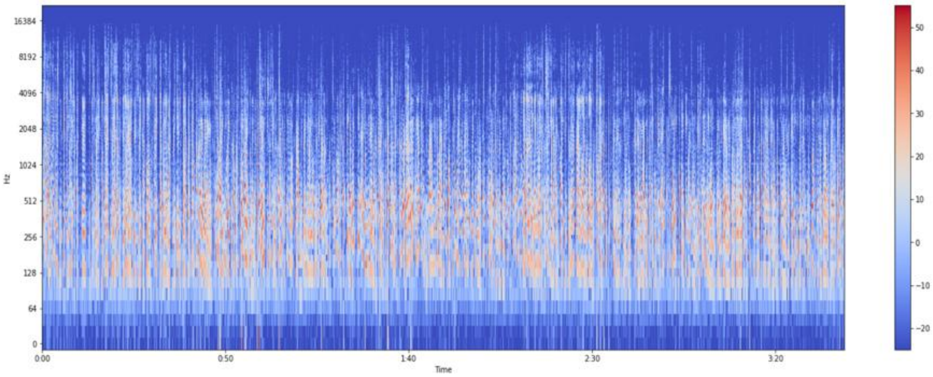


Figure 3 - Audio in Frequency Domain

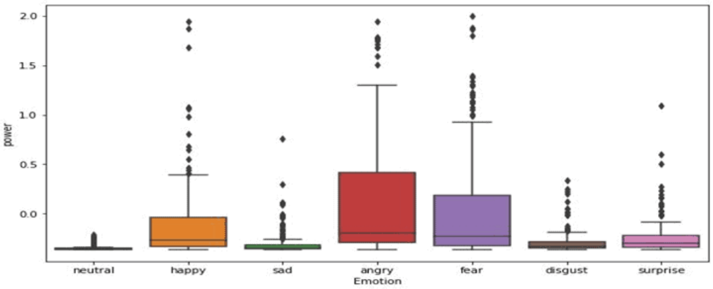


Figure 5 - Audio Energy Median Plot

The emotion extraction from the audio tone is done using a convolutional neural network (CNN) model. Sample data set for this pilot was obtained from TESS (Toronto Emotional Speech Set) , SAVEE (Surrey Audio-Visual Expression Emotion), and Berlin, that has a combination of male and female speakers, with the same words and sentences being pronounced in multiple emotions. Expression of emotions such as neutral, happy, sad, anger, fear, disgust, surprise are best estimated through the energy levels in the audio, or the power observed in the pitch itself. Emotions such as disgust, surprise or sadness, have very subdued tonal power and may be mostly expressed through facial expressions. On the other hand, emotions such as anger or fear is expressed through high energy dissipation. While this is intuitive, this data was trained through a machine learning classifier to predict dominant emotions in speech.

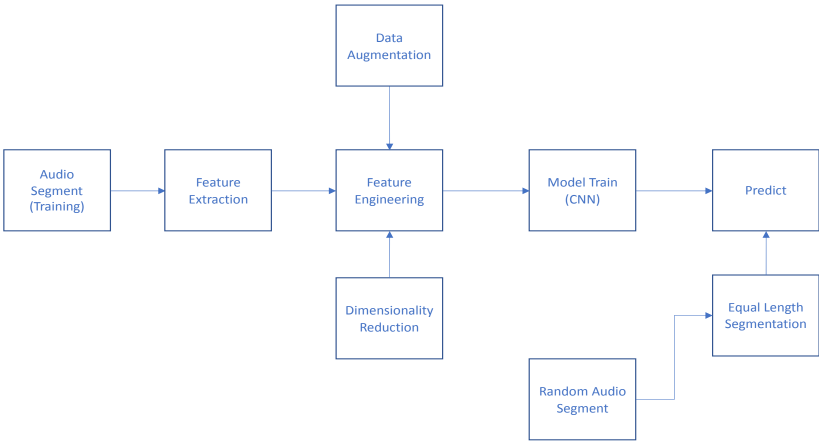


Figure 6 - Audio Emotion Detection Workflow

For each of the audio samples, feature extraction was done by obtaining the mel-spectrogram using Python Librosa package. The model was constructed in lines of VGG-16, but with some simplifications, given the nature of this pilot project. RELU was used as the activation function. The dense layers consisted of 512 units each, followed by dropout layers. SoftMax activation was used at the last output layer to get a probability distribution style output. Principal Component Analysis (PCA) was used for dimensionality reduction and to avoid overfitting of the model. The diagram below summarizes the overall training and classification approach.

Transcript Analysis – The audio segment is analyzed to generate transcription using Amazon Transcribe. The text that is obtained out of transcription is used for the following downstream analysis.

1. The corpus of text was analyzed for sentiment using Amazon Comprehend. Depending on the duration of the video (example: the transcript is more than 100 words), the transcript is first passed through a topic modelling to detect if there are distinct clusters identified within the text. It is possible that the content has evolved across the length of the video, and each such evolution has a different emotional tonality. For each segment, the text chunk is analyzed for the most dominant sentiment. The sentiment is reflected as a probability mix of positive, negative, or neutral. However, this sentiment classification cannot be very appropriately be fitted

into the circumplex model of emotions, as outlined above. To bridge this gap, a custom mood classification model was developed to better address the most dominant mood in the transcript.

2. A classifier based on logistic regression was developed for this pilot. The sample data set was created based on the publicly available data at https://paperswithcode.com/task/emotion-classification . The data was analyzed for multiple types of emotions such as joy, sadness, fear, surprise, anger, disgust, and shame. The data was cleaned for special characters, stop words, etc. was trained using SKlearn pipeline, based on the Logistic Regression model. The model predicts the probability distribution of different detection for an input text.
3. The pace of speech delivery and the variation thereof was computed, based on the duration of the video and the number of words in a unit time. Without using any machine learning, simple arithmetic metrics such as pace of speech delivery, if it is fast, slow, or normal, if there is a change in the pace of speech (may be because of an arousal) are captured. These metrics are important to determine the desired tempo of the music to be generated, that might blend as the best background music to the overall scene.

MUSIC GENERATION BASED ON EMOTIONS

With the dominant emotion of a video scene being detected, there are two fundamental ways to find the most appropriate music. This can either be done objectively, being driven by rules. Alternatively, generative artificial intelligence (AI) can be used to generate music based on a seed or a nominal key input. On the generative AI path, some of the popular models that can be explored are RNN, LSTM (recurrent neural network, long short-term memory), GAN (generative adversarial networks) or VAE (variational autoencoder).

The approaches outlined above have their own pros and cons. For example, music in itself does have the characteristic emotions attached to it. Musical segments played mostly in major notes tend to emit happy mood, while musical segments played in minor notes reflect a sad movement. Indian fine arts and music is composed of nine dominant emotions such as love, sorrow, laughter, anger, courage, fear, disgust, surprise and tranquility. There are ample number of Indian classical ragas reflecting most of these emotions. Each raga, has literally infinite number of movements and compositions, mapping further to different styles of tempo, rhythm and context. If this style is adopted for music mapping, there are a few challenges to be handled as well. These are:

- Rights availability of such music assets might be expensive and operation intensive.
- Given the sheer volume of content that is being produced today, there are chances to exhausting the uniqueness of content, and eventually being repetitive.
- It would require a high number of trained skills and effort, making the solution expensive.

Generative music is likely to have an endless variety, given the variation in the seed or being retrained from time to time. It is also possible to have a tighter control not only on the combination of nodes, but also on the rest, tempo, rhythm, pitch, so align better to the overall video segment. On a related note, and while still being on the early days of generative AI, there are questions related to data privacy and copyrights, such as labelled data used for training the models.

.

SOLUTION ARCHITECTURE

The diagram below outlines the end-to-end solution design from analyzing a video segment to assigning a customer background music to it based on the most dominant emotion.

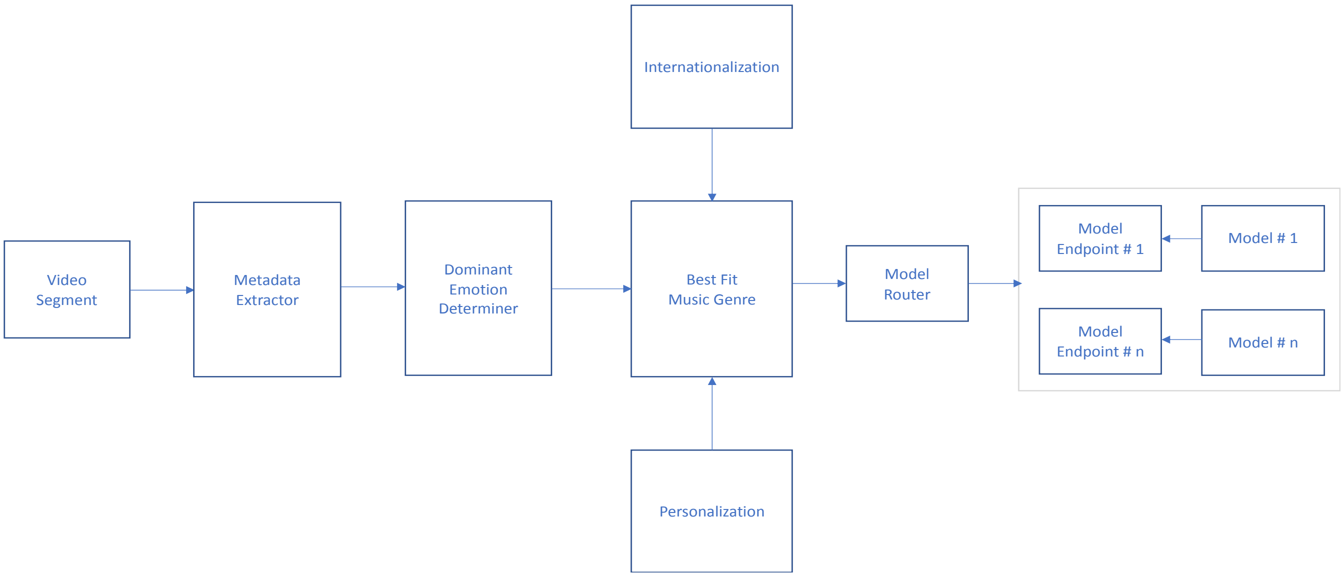


Figure 7 - High Level Solution Architecture

There are three variants of the use case that can be explored here. The first variant is based on the dominant emotion as detected from the video segment. This can be further enhanced based on internationalization aspects. For example, a globally popular content can be enjoyed with popular regional background music such as Arabic, African, or Indian. Similarly, and at an elevated cost of production, a viewer may specify the background score to be presented only on their instrument of choice such as violin and flute. Let us dive deeper into each of the key components of this solution.

Dominant Emotion Determiner – This is a rule-based component to determine the most dominant emotion in the scene. When the audio, video, and transcript are analyzed in parallel, each stream of metadata will produce a combination of emotions. In the current implementation, all such emotions are scored on a linear scale and added up to get the top three emotions. This might not be accurate in a production scale and for a wide genre of content. For example, for dramatized content the emotions may be portrayed through facial expressions, versus a court-room scene, a documentary or an action movie may be more heavily relying on the dialogs. It is therefore important to attach appropriate weight to the source of emotion (video, audio, transcript), based on the type of content.

Best Fit Music Genre – This is implemented as a machine learning classifier, using XGBoost. Since the data used for training and classification was (semi-)structured in nature, decent results were observed during the pilot phases, with a regular classifier. The labelled data can be sourced from third party music API, where either the mood and theme of the song is labelled out clearly or the dominant sentiment may be extracted from the lyrics. The second approach is through crowdsourcing a pool of musicologists to label a set of training data. The input to this classifier is the top three dominant emotions, ethnicity, pace (based on the speed of dialog delivery, or pace of scene change). The output to the classifier is a hyperlink to a seed melody, best fit music genre, and associated metadata.

Model Router – There are two fundamental approaches to train the melody generator model. The first approach is to create a large model with all genre of music being put together. The second approach is to create a separate model for every sub-genre of music, such as one for Indian classical, one for country music, one for jazz, etc. A model router is used to process the incoming request and route to the correct model endpoint. In the pilot conducted, even though with limited dataset, it was observed that the genre or the flavor of generated music remains purer, and sounds more realistic.

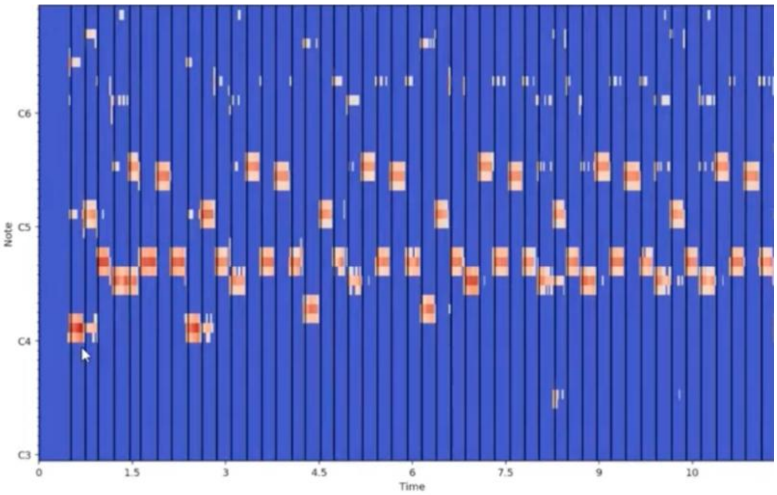
MACHINE LEARNING FOR MUSIC GENERATION

The following sections outline the key steps in designing a deep learning model to generate music based on a seed input melody.

Data Ingestion

Similar to any machine learning project, we would need labelled datasets to train the model. Needless to say, deep learning models such as Recurrent Neural Networks (RNN), Long Short Term Memory (LSTM) would not generalize and perform well unless there is a sufficient data volume for its training. There are three approaches that can be explored in this regard.

A. **Music Transcription** – Most of the music around us is present as audio files. While music transcription is a complex and evolving field, we explored a solution for a single musical instrument or vocal. As a part of this step, a music clip (example: MP3 / WAV) was analyzed and converted into a MIDI (musical instrument digital interface) format. This is a standard protocol to capture music information consisting of pitch, notes, timings, etc. into a computer understandable format.



Popular open-source libraries such as Librosa, music21, and midiutil are used for generating the music transcription. The key is to convert the audio clip from time domain to frequency domain using approaches such as FFT (Fast Fourier Transform), or CQT (Constant-Q Transform). Please refer to the appendix for a detailed discussion on the choice of CQT over FFT for this analysis. The diagram below illustrates a typical CQT plot as done for a sample music clip. This is a time-frequency representation with time on the X axis and the central musical frequency plotted on the Y axis (note). The black vertical lines illustrate what is called as a "onset" or the beginning of a note on a time space. It may be noted, that across each such onset block, there are multiple shades of yellow-orange. The deeper the color is, the more intense the pitch is. For generating MIDI scale for this pilot, the most dominant frequency was chosen in each of the onset blocks.

The diagram below illustrates the key steps in generating the midi file from the audio clip.

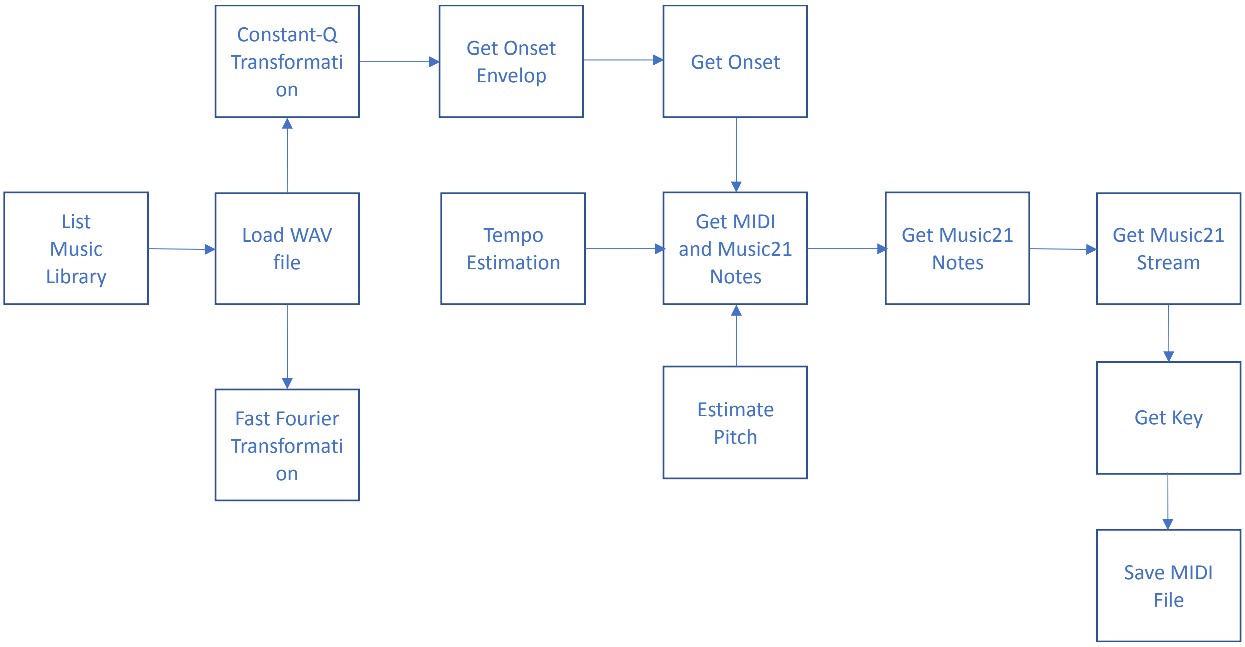


Figure 8 - Generate MIDI file from Audio

B. **Leverage Public Datasets** – Some of the popular and publicly available music datasets include the MusicNet, ESAC (Essen Associative Code and Folksong Database, with a collection of 5000+ songs all over the world), or similar others. Such datasets, while limited are reputed for a decent level of label accuracy and is used for training a pilot deep learning model for music generation.

C. **Human Notation –** Depending on the cost, criticality and time dimension, generating musical notations, transcriptions in the desired format such as MIDI, KERN, etc. can at least be a theoretical option. This is particularly true for niche musical effects, such as Arabic, African, or Asian music, that may not have abundance of labeled musical data in the first place.

Data Preprocessing

The core data pre-processing tasks are handled by open-source libraries such as Librosa, Keras, TensorFlow and Music21. However, it is important to understand the overarching design behind it. For this pilot, LSTM was chosen. Deep learning networks cannot handle string data as an input. A mechanism is required to convert the music information into a number representation.

Here is a sample melody track for reference. There are lots of information encode within it, such as notes, rest, pitch, key, notes per beat, octave and others.

The Music21 Python library is used to extract the key musical features. The y-axis on the MIDI notation represents the pitch, while the x-axis represents the time. In essence, the music notation is represented as a time series data. Every unit block of time (say every four beats), there is some musical activity as denoted by the note. Data pre-processing entails the key steps such as 1) Load the music file, 2) Detect the notes and rests, 3) Get the key directly from the song, or estimate the key, 4) Transpose major key to C-Major and minor key to A-minor. 5) Encode song into a time series notation, 6) Generate training sequences, 7) One-hot-encode the sequence. The diagram below summarizes the key steps:

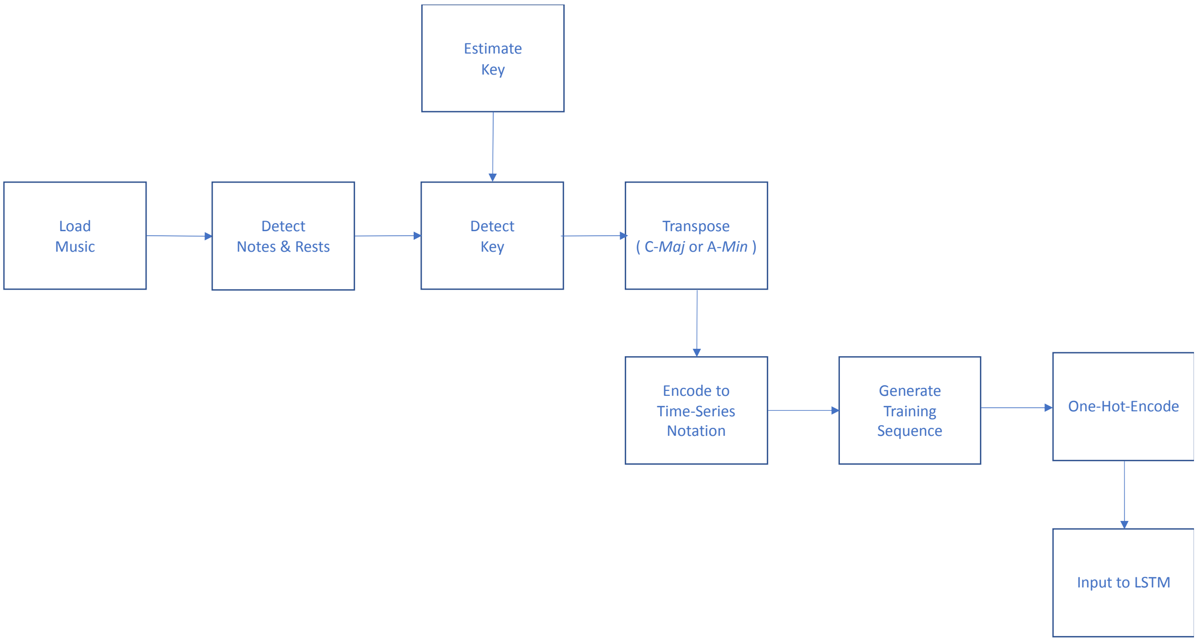
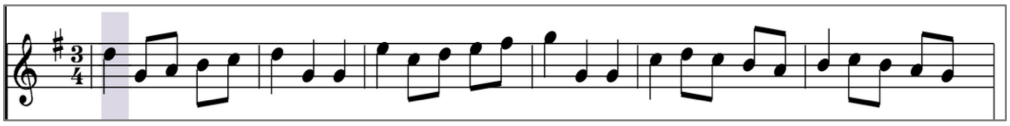

Estimate Key
Load Music
Detect Notes & Rests
Detect Key
Transpose (C-*Maj* or A-*Min*)
Encode to Time-Series Notation
Generate Training Sequence
One-Hot-Encode
Input to LSTM

Figure 9 - Data Preprocessing for LSTM Input

Model Train With LSTM

The model is built, compiled and trained using the Keras library. Following are the key characteristics of the RNN LSTM model:

- Number of output units – 38
- Number of input units – 256
- Learning rate – 0.001
- Epochs – 90
- Loss function – Sparse Categorical Cross Entropy



- Batch size – 64
- 20% Dropout
- Using Adam optimizer, with accuracy as output metric

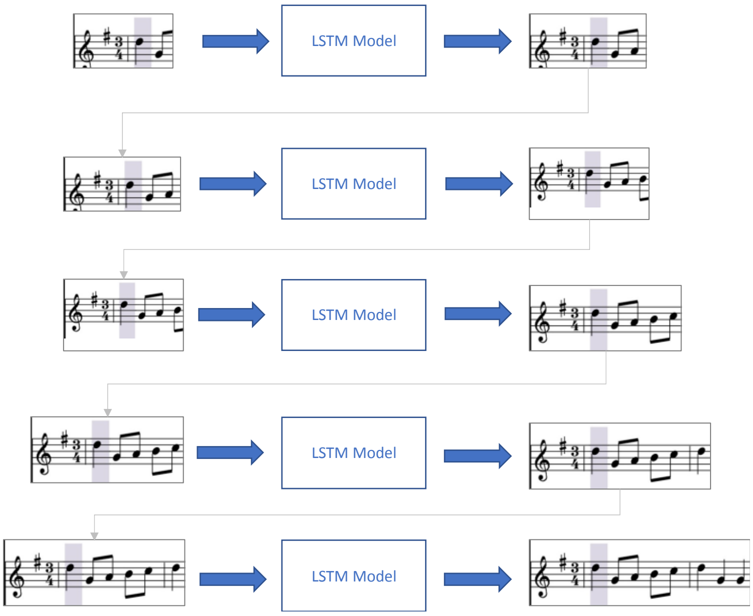


Figure 10 - Music Generation with LSTM

Generate Music with LSTM

The model trained in the previous step is used to generate music clip based on a given seed input. As is the characteristic with LSTM networks, the output from the network is keyed in as an input to the to the next call. When executed in a loop, this can generate a long music clip based on a short seed input. It may be noted that the seed note does include components of intended keys, pitch, and rest, such that the generated music clip is fit for the purpose of the video scene it is going to be used. The adjacent diagram illustrates how the music generation works at a high level. Apart from the melody input, other critical inputs to the model are number of steps to be iterated, maximum number of steps in the seed that should be used as an input, and a randomization factor to control how much probabilistic versus deterministic the output melody should be.

LSTM for Music

Recurrent Neural Networks (RNN) are types of neural networks that use output of the previous state as the input into the next state. In traditional neural networks, the inputs and outputs of one hidden cell is independent of that of the other cells. RNNs have a property of memory. The cells remember the structure of previous inputs and that is used to predict outputs. The classic challenges of vanishing or exploding gradient in RNN, is automatically avoided by adoption of LSTM.

There are a few traits of popular music segments such as returning back to the tonic note, at the end of a musical phrase, or using a popular combination of notes repetitively throughout the composition, or may be consistency of the rhythmic patterns. That is why context and memory of the music produced earlier is important. This is where the LSTM (long short-term memory) variant of RNN is critical since it can hold the long term and short -term memory of the just composed melody, and use it as a contextual memory to generate the subsequent musical phrases.

Model Train with GAN: Alternative Approach

While for this pilot LSTM was used to generate melody, Generative Adversarial Networks (GAN) is also a popular model to generate melody. The diagram below summarizes the key components of a typical GAN.

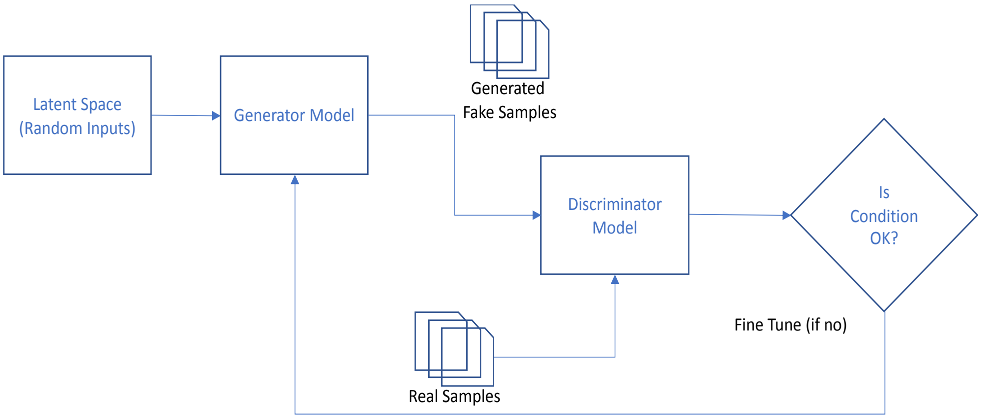


Figure 11 - Music Generation with GAN

The training input are pre-processed as usual into MIDI format. The generator and the discriminator models are foundationally convolutional neural networks in themselves. They are being bound together through the GAN, such that the loss of the discriminator can be back propagated into the generator. The MIDI files are converted to their corresponding image, representing the entire musical notation for that given music segment. With a good alignment between the generator and discriminator, this solution can generate music based on a seed. However, it may be noted that training and fine-tuning GAN models are tedious. For example, if the discriminator model gets trained faster and "expert" in detecting the fakes that are getting produced from the generator, the generator gets stuck and is not able to progress. The opposite situation is also true. If the discriminator is way too "lenient" and lets the fakes from the generator pass through – we would end up with a lot of output which does not truly reflective of the on-ground training data.

GAN models are more modern and sophisticated as compared to RNN and LSTMs of the world, but there is a cost associated in terms of the training effort. It is important to judge if the cost of training is justified given the business use case. Alternatively, pre-trained GAN models available as a service can be an easy way to start. Such solutions may have limitations in the extent of customization, as may be required while dealing with a niche genre of music.

CONCLUSION

In this paper we explored how we can generate background music for video segments using artificial intelligence. The bridge between the two worlds of video and music, is a common standardized set of emotional spectra expressed through circumplex models. It is more critical to capture the dominant sentiment from the video clip, being expressed through visuals, speech tone, speech content, etc. Music association can be simplified through rights acquisition of the right set of music catalogs, and have musicologists label the right set of emotions. This approach has a significant cost impact. Moreover, even if the emotion match, the pitch and tempo also have to blend. That is where machine learning comes in. With the capabilities in generative artificial intelligence, we can generate music in the matching pitch and tempo, based on a seed melody. The output MIDI file, can be played on an instrument of choice.

REFERENCES

1. Context-Dependent Domain Adversarial Neural Network for Multimodal Emotion Recognition (https://paperswithcode.com/paper/context-dependent-domain-adversarial-neural)
2. Multimodal Speech Emotion Recognition Using Audio and Text (https://paperswithcode.com/paper/context-dependent-domain-adversarial-neural)
3. Emotion Classification (https://en.wikipedia.org/wiki/Emotion_classification)
4. Emotion Detection and Recognition from text, using machine learning (https://devblogs.microsoft.com/cse/2015/11/29/emotion-detection-and-recognition-from-text-using-deep-learning/)
5. Music Information Retrieval (https://musicinformationretrieval.com/)
6. Amazon Rekognition (https://aws.amazon.com/rekognition/)
7. Amazon Transcribe (https://aws.amazon.com/transcribe/)
8. The Sound of AI (https://valeriovelardo.com/the-sound-of-ai/)

ACKNOWLEDGEMENTS

I would like to thank all my colleagues who have helped me to develop this concept, at its various stages of evolution. I would like to thank again all the contributors referred to in the reference section. I would also like to thank the International Broadcasting Convention (IBC) for publishing this paper.