International Journal of Data Science and Advanced Analytics (ISSN: 2563-4429)

Music Generation through Transformers

Shubhham Agarwal[1], Nailya Sultanova[2]
[1]Liverpool John Moores University, UK
[2] Kazan Federal University, Kazan, Russia
[1]shubhham.agarwal@gmail.com
[2]NRSultanova@kpfu.ru

Corresponding author email: shubhham.agarwal@gmail.com

***Abstract*— The content creator industry has been booming. After COVID-19 hit, the way content was observed changed drastically within the people using the social media platforms like Instagram, Facebook, Twitter, Tik Tok etc. There is an entire profession of "Influencers" who have become public figures due to their videos which are educational, or influencing the day to day decisions through reviews/unboxing or are just blogs/vlogs about passionate topics like traveling, food etc. This rise in video content has pushed the need for background music as well as the content creators are constantly adding in the videos with fresh, engaging music which sets the mood of the videos. True message is conveyed only when the music supporting the video matches with the mood of the viewers. The challenge is that music composition, editing etc. is a professional job and the influencers may not necessarily have those skills. Content creators can either try to create their own music which may have some transitional gaps or the quality of music may suffer or they can now buy the music through a licensing platform. In this research, transformers will be used to generate the transitional music for mixing two different sound clips.**

***Keywords*—music generation, transformer, AI composer, machine learning models and the deep learning models.**

1. Introduction

Influencer videos on different social media websites have seen a rise over the past few years. Videos ranging from tutorials to reviews to watch along to short reels/stories have dominated the platforms like Instagram, Tik Tok, YouTube etc. One of the major issues that these content creators are now facing is using a royalty free music which can be used across different platforms without facing the copyright claims on the music used in their videos. Since the audio mixing, composing is a professionals job, acquiring the basic background music for the videos becomes an expensive affair for the creators.

Over the period of 5 decades, researchers, developers, musicians, and technicians have been working on automating the music generation. Algorithms and programming has helped the researchers and composers think about the music as a process which can be processed as a set of instructions/grammar. These instructions can be stacked together to form a melody just like a sequence of notes. Steve Raich coined the term "Process Music" in 1968 where he described the music as a process and not as a process of compositions.

The algorithm based researches have been carried out since 1980. Briot introduced some of the concepts of deep learnings and related frameworks which explored the boundaries of music generation [1]. It further elaborated on the pros and cons of various different researches. Further there was Research in music and AI which reviewed different methods [2]. Various papers and researches talked about these methods however none of the papers went into the depth and provided relatively a high-level view of the algorithmic compositions [3][4]. Much of the motivations and needs to explore this field comes from the fact that as of now arts field is still undergoing the "AI transformation". While the technologies and methods have advanced over the past few years, the opportunities to scale this up and see better usage is still being discussed. Pearce et al. talked about the first problem statement that was conceived and how there is a motivation to create such algorithms [5].

Collins et al. talk about the music theory aspect and how it is being perceived by the artists who can see these evolutions coming in [6]. In 2013 David Fernández and Vico explored the AI methods in algorithmic compositions while explaining the early works and how they have evolved over the period [7]. Some of the methods have been covered in the sections below. The modern algorithms like ANN, RNN etc. have been discussed in the paper [8]. The functional aspect of music generation as a taxonomy has been reviewed in detail in the paper [9]. The paper investigates the functional aspects (melody, harmony, rhythm etc.) and how they are addressed in the algorithms. A recent paper Lopez-Rincon et al. looked into the classification of AI methods into different methods including Deep Learning architecture [10]. It is a derived version of the Briot's paper by the same author [11] . Widmer and Goebl, Kirke and Miranda investigated music performance and how the computational intelligence has been applied to this domain [12][13].

This study will aim to use different pre-processing techniques to develop relationship between the training songs and use that to generate cohesive performance music. This paper focuses on developing a basic AI composer for the content creators which they can use to generate the background music for their videos. This study will explore the traditional machine learning models and the deep learning models.

2. Materials and Methods

This research explores different techniques to generate music through different models. For any research and model development, it is necessary that the dataset is clean and is prepared as per the requirement. After that model building and

training work will start. Post that model evaluation will be investigated.

2.1. Dataset

MIDI - A MIDI (Musical Instrument Digital Interface) dataset is a collection of MIDI files, which are digital audio files that contain instructions for how to play a piece of music. MIDI files do not contain any actual audio data; instead, they contain a series of instructions that tell a device, such as a computer or synthesizer, how to play a piece of music. These instructions include information about what notes to play, when to play them, and how long to hold them.

Since these models are data intensive, this research would need relatively large dataset for training the model. For this purpose, the Lakh midi dataset which comprises of 176,581 midi files with all kind of songs out of which 45,129 songs have been matched and aligned to entries in the Million Songs dataset will be used.

In order to generate the music as per the user requirement, styles can handpick from the sites like https://freemidi.org. This will help in finetuning the model.

The other dataset that can be explored in this research is pop 909 dataset (https://paperswithcode.com/paper/pop909-a-pop-song-dataset-for-music) which is essentially a piano arrangement of the 909 Chinese pop songs. These songs are split into 3 tracks – one for the melody, one for the sub melody and one for the accompaniments (chords).

This research will make one key change i.e. changing the MIDI format to REMI format. The REMI format is a way of helping the model by adding this structure to the data. Instead of using exact start and end times the music is divided in bars, where every bar is divided in 16 parts to describe the starting points of the notes. For every note a duration is added. This duration has some more precision and can be described in 32th parts of a bar. Table 1 represents the differences in the MIDI representations and REMI representations.

Table1: Structural difference between MIDI and REMI representations [14]

Attribute	MIDI	REMI
Note Onset	Note on (0 – 127)	Same as MIDI
Note Offset	Note off (0 – 127)	Note Duration (32th note multiples; 1-64)
Time Grid	Time Shift (10-1000 ms)	Position (16 bins; 1-16) & bar(1)
Tempo Changes	Not defined	Tempo (30 -209 bpm)
Chord	Not defined	Chord (60 types)

As compared to the MIDI dataset REMI provides more track characteristics as compared to MIDI hence adding a layer of metrical structure to the MIDI dataset.

2.2. Data Pre-processing

For the ease of training, we will split the music into melody, bass and accompaniment. Salomon was working on an algorithm to sort the instruments in a MIDI track based on frequency of the notes, number of notes per minute, how many notes would sound at the same time, and some other metrics. This would make it possible to automatically pre-process the Lakh dataset.

Another thing the study will explore is the transposition of the songs. Since all the songs are in different keys it becomes a bit difficult to have a connection between the songs. There are 2 different ways to implement this:

• Using a python library called Music21 (music analysis library for python, and then transposing the song to the point where we get to the desired key)

• Transpose the songs to all keys however in this case the training would be 12 times the data as there are 12 different keys

This step ensures that the songs sound consistent for longer period.

2.3. Model Used

2.1.1. Transformers

The study will look at some of the transformer models which have shown promise in generating coherent music pieces for longer duration. As a basic structure, general transformers have 6 encoders and decoders. Fig. 1 below presents the transformer architecture. Each encoder has a multi head attention (self-attention) and position wise fully connected feed forward network while the decoder has 3 sub layers i.e. and position wise feed forward layer, multi head attention layer, and masked multi head attention layer. As the name suggests, the masked multi head attention ensures that the tokens are hidden and are not seen by the layers ahead before training.

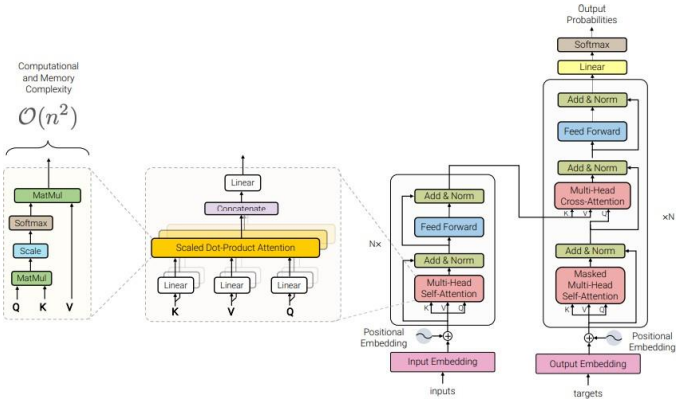

Fig.1 Illustrative Transformer model taken from [15]

After transformation of the MIDI and REMI files into musical characteristics, the sequence model will be implemented to predict the sequence of events.

One of the major factors for this decision is that the Transformers are able to generate notes without considering the influence of the following notes. The study assumes that it

will be an effective theory when used on Machine translation. However the challenge here with this approach is that the generated music as a whole may become a bit unstructured.

Transformer Wrapper and Autoregressive Wrapper are two techniques used in the Transformer architecture to improve its performance in different natural language processing (NLP) tasks.

A Transformer Wrapper is a technique used to modify the original Transformer model by adding new components to it. The added components are usually designed to improve the model's ability to perform a specific task.

On the other hand, an Autoregressive Wrapper is a technique used to modify the Transformer model's output by incorporating its own output as input in a recursive manner. This technique is particularly useful in tasks such as text generation, where the model generates a sequence of words or characters based on a given prompt or seed text. The Autoregressive Wrapper enables the model to generate sequences of arbitrary length by recursively predicting the next word or character based on the previous predictions.

Both techniques can improve the performance of basic Transformers in various NLP tasks. Transformer Wrappers can enable the model to learn better representations of the input text, while Autoregressive Wrappers can help the model generate more coherent and meaningful text.

2.1.1. **Evaluation**

Further analysis can be done by evaluating the features of the song. This study has looked at following features for analysis:

Chroma STFT depicts tonal content of the music. It indicates how much energy each of the pitch class (C, C#, D, D#, E, F, F#, G, G#, A, A#, B) is present in the track

• Roll Off helps identify the harmonics from the noise. It is the frequency under which some percentage of the total energy of the spectrum is contained

• Zero Crossing Rate is the rate at which a signal changes from positive to zero to negative or from negative to zero to positive. Its value has been widely used in both speech recognition and music information retrieval, being a key feature to classify percussive sounds

• Tempo refers to the pace of the music (can be fast or slow)

3. **Results and Discussion**

The study looked at following genres/bands/composers with some set of songs under each category. This section will look at the Audio Waves and spectrogram of the input music and the output music. This will be followed by the discussion around the quality of generated music and the scope to enhance it in the future

• Piano based pop songs – This category consists of 11 songs selected at random

• Classical Songs (Ludwig Van Beethoven and Wolfgang Amadeus Mozart) – This category consists of Piano Sonata No. 11 (Beethoven) also known as Opus 22 and Piano Sonata No. 10 (Mozart) composed on C major, K. 330/300h

• Pop/R&B (Coldplay) – This category consists of the three songs of the famous band Coldplay

• Rock (Audioslave) – This category consists of 3 songs classified as Rock songs sung by the band Audioslave

For the research we have taken single instrument as well as multi instrument songs to see how the output is impacted. Moreover the model has not been overloaded with songs of a specific genre as it was required to see how the model performs in case songs are similar or from one band/composer

For each of the output, following graphs will be created:

• Mel – Spectrogram: Mel spectrogram remaps the values in hertz to the Mel scale. The linear audio spectrogram is ideally suited for applications where all frequencies have equal importance, while mel spectrograms are better suited for applications that need to model human hearing perception

• Audio Waves: The generic term waveform means a graphical representation of the shape and form of a signal moving in a gaseous, liquid, or solid medium. For sound, the term describes a depiction of the pattern of sound pressure variation (or amplitude) in the time domain

For the conclusions, the AI generated tracks were shared with following personalities to get their perspective and evaluate the quality of the music generated. The feedback was collected from:

• Gaurav Dagaonkar: CEO GSharp Media, Composer/Singer

• Geet Sagar: Winner X-Factor India (2011), Singer/Composer

• Siddharth Sharma: Music Supervisor and Producer (Affiliated with Netflix).

The following section will look into the experiments that were run.

3.1 **Experiment 1**

While the AI generated output looks cohesive and has a melody as seen from the spectrogram and Audio Waves. Comparing the spectrograms of output with the spectrograms of the input, it can be observed that there are some gaps around 20 sec, 30 sec, and 50 sec mark in the AI generated track. However overall, this track is cohesive and is melodious

3.2 **Experiment 2**

Compared to the actual pieces the AI generated track does not really encapsulate the essence of the beautiful Sonata, however the track generated is very melodious and is cohesive. Building more advanced models on top of this with additional input will further enhance the quality of AI generated music. This generated music can have practical application in being used as a background music for YouTube videos, or short reels.

3.3 **Experiment 3**

Looking at the spectrogram and the audio waves of the original pieces (input), the AI generated output does not have the tempo, velocity of the input. This generated output is very different from the Mozart pieces however if the track is analysed as a standalone track, it is cohesive and melodious. An advanced model should be able to draw inspiration from the art of Mozart and create similar sounding harmony. While this track is not as melodious as the previous tracks (experiment 1 and 2), this still can be used in some situations and should pass an initial hearing test.

3.4 **Experiment 4**

The generated output has the same tune/melody as that of the first song – "A sky full of stars". It is likely that the rhythm/velocity of the music has overpowered the other songs as can be seen through the audio waves of the input songs. The opening of the "A sky full of stars" is much more powerful as compared to the other songs, however because of different instruments usage the AI generated track has a mix of different instrument with the melody of the first song. This track is not really AI generated rather looks inspired from the first track and is also not very melodious. The track would require human intervention to make it usable for actual real life application.

3.5 **Experiment 5**

In this AI generated track, there is not really a lot of melody or tempo as the tracks that were given as input had very different input sequences, melodies and tempo. The model is not able to generate any coherent sequence because of this and the output as expected is neither melodious nor cohesive. Although there is a 3-5 sec clip which is melodious and shows a promise of creating a potentially good song if we use advanced models and give an input of cohesive and similar sounding tracks.

4. **Conclusions**

The overall, as observed the single instrument, monotone music has produced a very nice melody which can pass as a human generated tracks and aligns closely with the objective of producing AI generated tracks with a fast turnaround and quality music

Furthermore, different song features like STFT, spectral roll off, zero crossing, harmony, percussion and tempo are extracted for the input tracks and the output tracks.

Table 2: Track feature extraction and comparison

Exp	Track Name (.mp3)	Chroma STFT	Roll Off	Zero Crossing Rate	Tempo
Exp 1	000	0.26955	3270.01709	0.07521	151.9991
	001	0.29986	3230.28217	0.06913	112.3471
	002	0.29085	3082.79001	0.06715	92.2852
	003	0.28900	3170.72753	0.07129	143.5547
	004	0.26552	3359.55352	0.07252	117.4538
	005	0.29218	3642.19852	0.08937	99.3840
	006	0.28035	3174.40672	0.06774	123.0469
	007	0.31286	3205.33881	0.07078	95.7031
	008	0.27737	3150.26202	0.06634	151.9991
	009	0.25158	3549.00292	0.08737	161.4990
	010.mp3	0.28171	3166.27452	0.07164	112.3471
	Output_pop_20.mp3	0.33923	2914.52932	0.06079	123.0469
Exp 2	beethoven_opus22_1.mp3	0.28344	3711.62658	0.08617	161.4990
	beethoven_opus22_2.mp3	0.25736	3414.73368	0.07831	129.1992
	beethoven_opus22_3.mp3	0.26442	3821.11108	0.08853	123.0469
	beethoven_opus22_4.mp3	0.27393	3837.82832	0.09242	172.2656
	Output_beethoven_opus_22.mp3	0.28415	3482.99880	0.07674	123.0469
Exp 3	mz_330_1.mp3	0.25790	4168.89540	0.11419	135.9992
	mz_330_2.mp3	0.22399	3341.43590	0.08068	92.2852
	mz_330_3.mp3	0.27273	4008.18073	0.10759	172.2656
	output_mz_330.mp3	0.26493	3427.12760	0.07677	129.1992
Exp 4	ColdplayIn_My_Place.mp3	0.36340	4440.68699	0.12043	143.5547
	coldplaythe_scientist.mp3	0.35152	3366.17697	0.08479	143.5547
	Coldplay_A_Sky_Full_of_Stars.mp3	0.29231	3563.41859	0.07948	117.4538
	output_coldplay.mp3	0.37264	2765.58645	0.05519	117.4538
Exp 5	BeYourself.mp3	0.39569	3925.54730	0.14089	117.4538
	LikeAStone.mp3	0.34115	5248.18685	0.12529	112.3471
	SevenNationArmy.mp3	0.41222	4379.61567	0.12846	123.0469
	output_audioslave.mp3	0.52276	3979.43242	0.14157	117.4538

In the above table, the highlighted tracks represents the features of the output tracks as compared to the input tracks under each experiment that has been conducted.

Except for the generated track for Audioslave, all the other generated tracks are in line or has similar number as compared to their input tracks. The roll off frequency for the tracks are generally in line with the input tracks for the generated tracks. Output from experiment 4 indicates a lower roll off frequency as compared to the inputs.

All the generated tracks have the tempo marking between 105-132, which is also called Allegro type. Some if the input tracks have a higher tempo which can be classified as "Allegrissimo" – (160-184 bpm) and "Presto" (168 - 200 bpm).

This study explored the use of Autoregressive wrappers and Transformer wrappers to enhance the attention of the model on the input. This study further identified how converting the MIDI dataset to REMI dataset gives and extra detailed information to the model to help generate better songs. These

results can be used in the future to further enhance the model by using a combination of Autoregressive wrapper with the continuous transformer wrapper to help in better decision making to produce high quality music or even complete a song.

Identifying the right aesthetics, tempo and velocity of the music will be key criteria in generating beautiful compositions. As observed in this research, Piano pop song is able to generate a hummable melody which can pass a composition produced by a beginner to intermediate piano player, with more sophisticated models which can have hyperparameters tuned as per the genre of the model can create and generate good music for the mass audiences.

X-transformers have shown a way to produce quick results without relying on the pre-trained data and is cost-effective. Further there are multiple ways by which features of the songs/ tracks can be extracted. We have used MFCC features for understanding the track through statistics, however there are multiple other ways to extract the feature of the songs:

• BFCC which is Barc frequence cepstral coefficients which uses bark scales. The bands of frequency in Bark Scale are almost linear to 500 Hz

• NGCC uses the Gammachirp filters followed by the logarithmic power spectrum

• GTCC uses the Gammatone cepstral coefficients are different set of features extracted from FFT based techniques

• LFCC uses the linear filter bank that is dynamic and more robust as compared to MFCC in babble noise

References

[1] J.-P. Briot, “From Artificial Neural Networks to Deep Learning for Music Generation”. History, Concepts and Trends, 2020. [Online]. Available: http://arxiv.org/abs/2004.03586.

[2] C. Roads (n.d.) “Research in Music and Artificial Intelligence”.

[3] G. Loy and C. Abbott, (n.d.) “Programming Languages for Computer Music Synthesis, Performance, and Composition”.

[4] G.Papadopoulos and G. Wiggins, (n.d.) “AI Methods for Algorithmic Composition: A Survey, a Critical View and Future Prospects”.

[5] M. Pearce, D.Meredith and G. Wiggins, (n.d.) “Motivations and Methodologies for Automation of the Compositional Process”.

[6] N. Collins and A.R. Brown, “Generative music editorial. Contemporary Music Review”, 2009.

[7] J. David Fernández and F. Vico, “AI Methods in Algorithmic Composition: A Comprehensive Survey”. Journal of Artificial Intelligence Research, vol. 48, pp.513–582, 2013. [Online] Available: http://www.flexatone.net/algoNet/ [Accessed 26 Oct. 2022].

[8] C.H. Liu and C.K. Ting, “Computational Intelligence in Music Composition: A Survey”. IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 11, pp.2–15, 2017.

[9] D. Herremans, C.H. Chuan and E. Chew, “A functional taxonomy of music generation systems. ACM Computing Surveys”, 505, 2017.

[10] O. Lopez-Rincon, O. Starostenko and G.A.S. Martin, “Algoritmic music composition based on artificial intelligence: A survey”. 2018 28th International Conference on Electronics, Communications and Computers, CONIELECOMP 2018. Institute of Electrical and Electronics Engineers Inc., pp.187–193, 2018.

[11] J.-P. Briot, G. Hadjeres and F.-D. Pachet, “Deep Learning Techniques for Music Generation -- A Survey”, 2017. [Online] Available: http://arxiv.org/abs/1709.01620.

[12] G. Widmer and W. Goebl, “Computational models of expressive music performance: The state of the art”. Journal of New Music Research, vol. 333, pp.203–216, 2004.

[13] A. Kirke and E.R. Miranda, “An Overview of Computer Systems for Expressive Music Performance”. Guide to Computing for Expressive Music Performance. Springer London, pp.1–47, 2013.

[14] J.L. Hsu and S.J. Chang, “Generating music transition by using a transformer-based model”. Electronics (Switzerland), p. 1018, 2021.

[15] Y. Tay, G. Research, M. Dehghani, D. Bahri, and D. Metzler, (n.d.) “Efficient Transformers: A Survey”.