Machine Learning for Music Generation

A Seminar Report

Submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of the requirements for the award of the degree

Master of Computer Application

by

JAGAN S NAIR

CHN23MCA-2037

Department of Computer Engineering

College of Engineering Chengannur Kerala - 689121

Phone: (0479) 24541125 Fax: (0479) 2451424

Website: www.ceconline.edu

JANUARY 2025

DEPARTMENT OF COMPUTER ENGINEERING

COLLEGE OF ENGINEERING CHENGANNUR

2024-2025

CERTIFICATE

This is to certify that the **20MCA244 SEMINAR** report titled **"Machine Learning for Music Generation"** submitted by JAGAN S NAIR (CHN23MCA-2037) to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the MCA degree is a bonafide record of the seminar work carried out by him under our guidance and supervision. This report, in any form, has not been submitted to any other University or Institute for any purpose.

Smt.Linnet Elsa John
(Seminar Guide)
Assistant Professor,
Dept. of Computer Engineering
College of Engineering,
Chengannur

Smt.Shereena Thampi
(Seminar Coordinator)
Assistant Professor,
Dept. of Computer Engineering
College of Engineering,
Chengannur

Sri.Gopakumar G
(Associate Professor)
Head of the Department,
Dept. of Computer Engineering
College of Engineering,
Chengannur

Acknowledgement

This work would not have been possible without the support of many people. First and foremost, I give thanks to Almighty God who gave me the inner strength, resources, and ability to complete my seminar successfully.

I would like to express my gratitude to **Dr. Hari VS**, The Principal, for providing the best facilities and atmosphere for the seminar completion and presentation. I also extend my sincere thanks to the Head of the Department **Sri.Gopakumar G** (Associate Professor, Computer Science and Engineering), our seminar coordinator **Smt. Shereena Thampi.**(Assistant Professor, Computer Science and Engineering), our seminar guide **Smt.Linnet Elsa John** (Assistant Professor, Computer Science and Engineering) for their invaluable guidance, encouragement, and support throughout this endeavor.

I would like to give proper credit to the authors of **Enhancing Music Genre Classification with Artificial Intelligence**, which was the main reference material I used for this seminar. Most images in this seminar report are from the afore mentioned research article.

JAGAN S NAIR

Abstract

Delves into advanced music generation using hybrid models that combine deep neural networks, machine learning algorithms, variational autoencoders (VAEs), long short-term memory (LSTM) networks, and Transformers to craft diverse and engaging musical experiences. The research aims to deepen the understanding of music's influence on human lives while developing methodologies to tailor compositions to individual preferences. By extracting features such as spectral properties, rhythmic patterns, and tonal characteristics from a large and diverse music collection spanning multiple genres, the study establishes a robust foundation for generation models.The music generation process leverages the unique strengths of VAEs, LSTMs, and Transformers.VAEs facilitate the creation of novel compositions within specified genres by learning a continuous latent space representation of the samples.LSTMs and Transformers excel at modeling temporal dependencies and the intricate patterns inherent in musical structures. Although the approach does not claim state-of-the-art performance, the results highlight its potential in enhancing music-related applications, such as personalized recommendation systems and creative tools for composers.

Contents

List of Figures

1. Introduction

The transformative power of music has been a cornerstone of human culture and expression throughout history. As diverse as our societies, the vast array of music genres reflects our rich cultural heritage and individual creativity. With the rapid advancements in technology, new paradigms in music analysis and synthesis are emerging, further enhancing our understanding of this universal language.

This chapter introduces the challenges and opportunities associated with music generation using machine learning and deep neural networks. It focuses on leveraging hybrid models such as Variational Autoencoders (VAEs), Long Short-Term Memory (LSTM) networks, and Transformers to create diverse and engaging musical experiences. These advanced models help replicate the artistic complexities of music while ensuring efficient generation across various genres. [5]

Music generation can be optimized through methods like VAE-based representations that provide a continuous latent space for the generation of novel compositions within a genre. Additionally, deep learning models like LSTMs and Transformers are used to capture temporal dependencies and intricate patterns in music. The integration of these models allows for both creativity and coherence in AI-generated music, aiming to match the emotional resonance and thematic consistency of human-produced music.

Furthermore, introduces the potential use of AI-driven models in real-world applications such as music recommendation systems, personalized music generation, and creative tools for composers. Through improved training methods and better dataset handling, the feasibility of generating high-quality compositions, including complex musical patterns, is significantly enhanced.

The goal is to explore and expand the possibilities of AI in music generation while addressing the key challenges of creativity and diversity to genre.

- **Phase 1: Dataset Collection and Preprocessing**
  - Collect and preprocess large, diverse music datasets for genre classification and generation tasks.
  - Extract relevant features from music samples, including spectral properties, rhythmic patterns, and tonal characteristics.
  - Develop effective feature extraction techniques to enhance model performance.
- **Phase 2: Model Development and Training**
  - Implement hybrid models that combine VAEs, LSTMs, and Transformers for music generation tasks.
  - Train the models on various music genres, optimizing for creativity and coherence.
  - Explore different architectures and hyperparameter tuning for optimal performance.
- **Phase 3: Evaluation and Testing**
  - Evaluate the generated music using subjective and objective metrics, including musicality and genre fidelity.
  - Conduct user studies to gauge listener satisfaction and identify areas for improvement.
  - Benchmark model performance against state-of-the-art music generation systems.
- **Phase 4: Application in Real-World Systems**
  - Deploy trained models in music-related applications, such as personalized music generators and recommendation systems.
  - Explore the integration of AI-driven models with existing music production tools to assist composers.
  - Expand the scope of music generation to more complex genres and styles.

- **Phase 5: Continuous Improvement and Future Directions**
  - Enhance model architectures to further improve the quality of generated music.
  - Investigate the use of unsupervised and reinforcement learning techniques for more creative compositions.
  - Explore new datasets and music genres to broaden the applicability of music generation models.

1.1 Research Contributions

1. **Hybrid Models for Music Generation**

Recent advancements in music generation have shown the power of hybrid models that combine VAEs, LSTMs, and Transformers. VAEs help capture latent representations of music, which are crucial for generating novel compositions. LSTMs and Transformers are used to model the sequential nature of music, handling long-term dependencies across notes and chords. These models ensure that the generated music not only respects the structure of existing genres but also brings creativity to the table, producing new compositions that align with human musical tastes.

2. **Improved Dataset Handling and Feature Extraction**

The research explores various datasets, such as GTZAN, Free Music Archive (FMA), and the Million Song Dataset (MSD), which provide rich sources of music data for training. The feature extraction techniques applied to these datasets—ranging from spectral analysis to rhythm and tonal characteristics—allow for more accurate and efficient model training. This approach enhances the models' understanding of different genres, improving their ability to generate music that is stylistically consistent and emotionally resonant.

2. Literature Review

2.1 StemGen: A Music Generation Model That Listens (2024)

StemGen, introduced by Parker et al. in 2024, is a transformer-based music generation model designed to respond actively to musical context [1]. This model stands out for its ability to generate high-quality audio that maintains strong musical coherence with the given context, allowing for compositions that feel natural and musically consistent. By leveraging the power of transformer architectures, StemGen can produce music that fits seamlessly with melodies, rhythms, or even partial compositions, enhancing its applicability in diverse fields like film scoring and video game music. Its ability to produce coherent music is a significant step forward compared to previous AI-generated compositions, which often lacked structure or seemed disconnected from the context. This makes it a promising tool for creative applications where dynamic and responsive music is required.

However, the model is not without its limitations. It demands significant computational resources, making it challenging to run on less powerful systems or for individuals with limited access to high-end hardware. Furthermore, while StemGen excels in generating contextually appropriate music, it offers limited control over fine-grained details, such as specific instrument articulation or intricate stylistic elements, which can be crucial for more specialized or highly personalized compositions. These limitations mean that, while the model is an impressive step forward in AI-driven music generation, it still requires refinement to provide the level of customization and flexibility that some creative users might desire.

2.2 Graph-based Polyphonic Multitrack Music Generation (2023)

In 2023, Cosenza et al. introduced a groundbreaking method for polyphonic multitrack music generation by combining a graph-based representation of music with a deep Variational Autoencoder (VAE). This innovative approach utilizes graphs to represent the relationships between different musical elements, such as notes, chords, rhythms, and instruments [2]. By modeling music in this way, the system can generate complex, multitrack compositions that preserve tonal and rhythmic consistency across different layers of music, ensuring that harmonies and rhythms complement each other effectively. The use of a deep VAE allows for the generation of diverse musical graphs that can be conditioned on specific instruments, enabling the model to generate music that mimics the sound and structure of individual instruments or a full orchestra, depending on the input. This flexibility makes the method highly valuable for tasks like generating realistic multitrack music, where coordination between various instruments is key.

Despite its advantages, the approach faces a number of challenges. One of the primary difficulties is the complexity involved in modeling intricate musical structures, such as counterpoint, complex harmonies, and detailed orchestration, which are often difficult to represent accurately in a graph-based format. While the method excels at maintaining consistency in more straightforward compositions, it can struggle with capturing the nuanced relationships and sophisticated musical forms found in more advanced compositions. Additionally, the model requires a large amount of high-quality training data to learn the relationships between musical elements effectively. The need for extensive datasets can be a limitation, especially in domains where large, well-labeled music collections are not readily available. Overcoming these challenges could further enhance the model's ability to generate highly sophisticated and stylistically diverse music while reducing its dependence on vast amounts of training data.

2.3 JEN-1: Text-Guided Universal Music Generation with Omnidirectional Diffusion Models (2023)

In 2023, Li et al. introduced JEN-1, a model designed to generate music from text descriptions using a novel approach that combines both autoregressive and non-autoregressive training methods [3]. By leveraging these two distinct training techniques, JEN-1 aims to capture the advantages of each, offering a flexible and computationally efficient solution for text-to-music generation. Autoregressive models are effective for sequentially generating content, ensuring that the music flows coherently over time, while non-autoregressive methods can speed up the generation process by making predictions in parallel. This hybrid approach allows JEN-1 to produce high-quality, musically coherent pieces while reducing the computational load compared to purely autoregressive models. The model can take textual prompts—describing emotions, genres, or specific musical attributes—and translate them into full musical compositions, offering a promising tool for applications like soundtrack generation, automated composition, or even creative exploration.

Despite its strengths, JEN-1 faces challenges, particularly in its ability to capture complex musical nuances. While it can generate music that is tonally coherent and musically appropriate, it may struggle to model the more intricate elements of music, such as advanced harmonic progressions, subtle dynamics, or highly detailed orchestration, especially when trained on large and diverse datasets. These finer aspects of musical composition can be difficult for any AI model to grasp, and while JEN-1 is capable of producing high-quality output in many cases, it might lack the depth and richness of music composed by human experts. Additionally, the scale of the dataset required for training, which includes a broad range of musical styles and textual descriptions, could pose difficulties in terms of generalization and fine-tuning, leading to potential inconsistencies or oversimplifications in the generated music. As with many generative models, continued improvements in both training techniques and data quality will be key to addressing these limitations and enhancing the model's ability to generate more nuanced and sophisticated musical compositions.

2.4 Progressive Distillation Diffusion for Raw Music Generation (2023)

In 2023, Pavlova introduced a novel approach for raw music generation through a diffusion model, specifically utilizing a 1D U-Net architecture. Diffusion models, which have gained prominence in generative tasks, work by progressively refining random noise into structured output through a series of steps. In Pavlova's implementation, the diffusion process is applied to raw audio files, enabling the model to generate high-fidelity audio directly, without relying on pre-processed representations like spectrograms or symbolic music formats. The use of a 1D U-Net architecture—an established model in image generation tasks—leverages its ability to capture spatial hierarchies and details in the audio signal, making it well-suited for the task of generating coherent and realistic sound waves. This approach is adaptable to a wide range of music genres, allowing for diverse musical compositions that maintain high-quality audio production, making it useful in creative industries such as film scoring, game music, and music production [4].

However, the model comes with notable challenges, particularly in terms of computational efficiency. Diffusion models require numerous steps to refine the generated audio, which can be computationally intensive, especially for long compositions or when scaling the model for large datasets. This translates into longer processing times and higher demands on hardware resources, which can make the model less accessible for real-time applications or for users with limited computational power. Additionally, while the high-quality audio generation is a major advantage, the slower speed of generation poses a significant drawback when compared to other generative models, such as autoregressive models or GAN-based approaches, which can produce results more quickly. The trade-off between high fidelity and generation speed means that, while the model excels in audio quality, it may not be ideal for applications where real-time or rapid music generation is crucial. Continued optimization of the diffusion process and model architecture could help address these performance concerns in future iterations.

3. EXISTING SYSTEM

3.1 Related Work

3.1.1 JukeBox

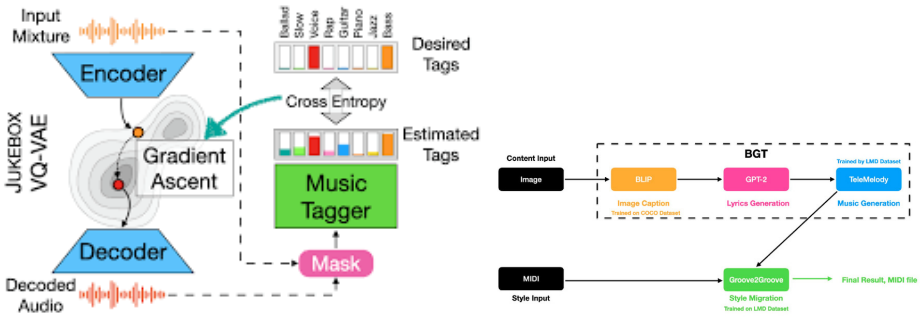
A generative model called JukeBox, which can create music and singing in the raw audio domain. It uses a multi-scale VQ-VAE (Vector Quantized Variational AutoEncoder) to compress raw audio to discrete codes and models those using autoregressive Transformers. This model can be conditioned on the artist and genre to steer the musical style. It was trained on a large dataset of 1 million songs, paired with the corresponding metadata. After a slow training process due to the complexity of the model, the results shown are quite remarkable. It generates coherent music pieces with harmony, rhythm, and even singing in multiple genres. Even if the model has its limitations in controlling the high-level attributes of the generated song, its capabilities in mimicking artist's styles and generating lyrics are still impressive.

3.1.2 Groove2Groove

One-shot transfer, which involves taking a piece of music in one style (e.g., jazz) and transforming it into another style (e.g., rock), using only a single example of the target style. The model consists of a style encoder, which takes as input a single example of the target and encodes it, and a decoder which takes as input a MIDI file and the target style and outputs a new MIDI file containing the original content in the target style. The model is evaluated on a variety of musical styles (not only broad genres) and shows that it can perform the transfer with high fidelity. This approach has the potential to be very useful in a variety of musical applications like remixing music.

3.1.3 ThemeTransformer: Symbolic Music Generation with Theme-Conditioned Transformer

A novel approach for generating symbolic music by conditioning a Transformer model on thematic material, rather than using the more common prompt-based conditioning. The theme-based conditioning ensures that the theme repeats and varies appropriately throughout the generated piece. To achieve this, the authors developed a technique to automatically retrieve thematic material from music pieces using contrastive representation learning and clustering. This method segments music into fragments, clusters them to identify recurring themes, and selects representative fragments as the thematic condition. Regarding its performance, the model was evaluated both objectively and subjectively against traditional prompt-based models. The results showed that the Theme Transformer can generate polyphonic pop piano music that better incorporates and varies the thematic material, offering more musically coherent and interesting compositions compared to previous baseline models.



(a) Jukebox (b) Groove2Groove

(c) Theme Transformer

Figure 3.1: Jukebox,Groove2Groove and Theme Transformer systems

3.2 Existing Applications

Spotify is a popular digital music, podcast, and video streaming service that gives access to millions of songs and other content from creators all over the world []. Artificial Intelligence (AI) plays an integral role in Spotify's functionality, analyzing the acoustic, cultural, and personal inputs for every user to create personalized recommendations. There are even some AI-driven systems that predict upcoming hits based on users' behavior patterns. Moreover, Spotify is developing several models to create music and provide songwriting assistance, showcasing AI's potential in the creative aspects of the music industry. [5]

Figure 3.2: Spotify's AI-Driven Music Recommendation System

SoundHound is a versatile and popular music application that serves as a platform that identifies songs, displays real-time lyrics, and even offers a voice-controlled AI. The functioning of SoundHound's song identification system is based on a methodology known as audio fingerprinting. A user plays a song near the device running SoundHound, the app listens to a segment of a song, transforms it into a unique numerical identifier (fingerprint), and matches it against a vast database of music [5]. The process is very fast and can identify songs within a matter of seconds, even in noisy environments. Furthermore, it has a unique ability to identify a fingerprint even on a user's humming, singing, or whistling, without the need for the original recording or lyrics.

Figure 3.3: SoundHound

Soundraw is an AI-driven platform designed to democratize the music production process. The app enables users to generate original music compositions based on various attributes, accessible to both novices and musicians alike [7]. One of the most impressive features is the ability to create music according to a genre. The result can also be fine-tuned to reflect a desired mood or a theme, leading to highly personalized pieces of music tailored to users' preferences. Other powerful features include tempo and length altering and also audio editing tools like specifying notes, changing instruments, or adjusting the mix of the track.

Figure 3.4: Soundraw's AI Music Composition

4. PROPOSED SYSTEM

4.1 Proposed Solution

4.1.1 Music Notation

Music notation serves as a visual coding system to depict music, using a set of symbols. Each music notation is different and is used in its way. For example, the tablature notation is instrument-specific and provides information about the physical placement of the performer's fingers. The tab consists of horizontal lines that represent the strings of the instrument and numbers that represent the fret where the fingers should be placed. It can also indicate techniques such as slides, harmonics, or vibrato.

Despite the effectiveness of many traditional forms of musical notation for live performances, they have visible limitations when it comes to playing with music in the digital realm. Here, the Musical Instrument Digital Interface (MIDI) excels [5]. MIDI is a protocol that enables computers and musical instruments to communicate with each other. It does not contain any sound, as it encodes information about the audio track, like the pitch, velocity, vibrato, or volume. The nature of this notation allows for easy transposition and changing time signatures or instruments (e.g., the piano part can be easily switched with any other instrument in the library). MIDI is also very efficient and economical: a symphony that might require hundreds of traditional sheet pages can be stored in a lightweight file, making it perfect to work with for generation tasks.

Octave	Note numbers											
	Do	Do#	Re	Re#	Mi	Fa	Fa#	Sol	Sol#	La	La#	Si
	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
0	0	1	2	3	4	5	6	7	8	9	10	11
1	12	13	14	15	16	17	18	19	20	21	22	23
2	24	25	26	27	28	29	30	31	32	33	34	35
3	36	37	38	39	40	41	42	43	44	45	46	47
4	48	49	50	51	52	53	54	55	56	57	58	59
5	60	61	62	63	64	65	66	67	68	69	70	71
6	72	73	74	75	76	77	78	79	80	81	82	83
7	84	85	86	87	88	89	90	91	92	93	94	95
8	96	97	98	99	100	101	102	103	104	105	106	107
9	108	109	110	111	112	113	114	115	116	117	118	119
10	120	121	122	123	124	125	126	127				

Figure 4.1: MIDI Datasets

4.1.2 Datasets

When seeking the perfect dataset for the music genre classification task, it is essential to consider the size of the dataset, its diversity, and the variety of genres it covers. Datasets with varied musical styles are crucial to help train models that can generate or classify music across different genres. Datasets used for this task often include labeled music tracks with genre-specific tags [5]. These datasets are often composed of audio files, but in some cases, they may also include symbolic music formats like MIDI.

1. GTZAN Dataset

- **Overview**: Often the first choice for music genre recognition tasks due to its simplicity, small size, and balanced structure.
- **Content**:
  - 1,000 audio tracks, evenly distributed across 10 popular genres.
  - Each genre contains 100 30-second audio tracks stored in labeled subfolders.
  - A second folder contains Mel Spectrograms of the audio files, also distributed evenly across the 10 genres.

- **Advantages**:
  - Balanced genre distribution eliminates the need for preprocessing to address class imbalance.

2. Free Music Archive (FMA) Dataset

- **Overview**: A richly annotated and diverse collection of music for various music analysis tasks, including genre tagging.
- **Content**:
  - Tracks are organized hierarchically into 16 top-level genres (e.g., Pop, Rock, Electronic) and 161 sub-genres (e.g., Psych-Rock, Lo-Fi, Drone).
  - Each track includes metadata like ID, title, album, release year, genres, and sub-genres.
- **Advantages**:
  - Detailed annotations and hierarchical taxonomy support a broad range of tasks.

3. Million Song Dataset (MSD)

- **Overview**: A large-scale collection of audio features and metadata for a million contemporary music tracks.
- **Content**:
  - Metadata includes release year, artist, popularity, key, tempo, duration, and more.
  - Features like rhythm, timbre, segments, bars, and beats computed using The Echo Nest API.
- **Advantages**:
  - Massive size offers extensive data for research in music analysis.

- **Limitations**:
  - Does not include actual audio files due to copyright restrictions.
  - Heavily biased toward popular Western music, limiting its diversity and representation of world genres.

4. Spotify API

- **Overview**: A rich source of music data with access to track details, artist information, album data, playlists, and audio analysis.
- **Features**:
  - Provides detailed audio features like tempo, key, danceability, energy, and more, along with metadata.

The size and diversity of these datasets provide a strong foundation for training machine learning models to generate or classify music in specific genres effectively.

4.2 Music Generation Processes

4.2.1 Dataset Selection

The initial step in generating music in a specific style is obtaining a representative dataset. Among 10 popular genres (blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, rock), jazz stood out due to its improvisational nature, making it ideal for AI to "improvise" within defined musical parameters. Jazz's emotional depth, expressed through instruments, allows AI to replicate patterns evoking specific feelings. A suitable jazz dataset in MIDI format, containing 935 files and pre-computed properties in CSV, is available.

4.2.2 Data Preprocessing

Audio data, transformed into text-like formats, simplifies processing. Notes and chords are extracted and saved as text for efficiency. Each sequence of 100 notes predicts the

next note, enabling melody generation by shifting and repeating sequences until the desired track length is achieved.

4.2.3 Deep Learning Methodology

To enhance model robustness, data augmentation was applied, creating 4 mel spectrograms per track: unmodified, frequency-masked, time-masked, and noise-added (mean 0, SD 0.3). This process expanded the dataset to 32,000 files evenly distributed across 10 folders.

4.2.4 Machine Learning Techniques

Two CSV files were created, containing mean and variance of features (e.g., zero-crossing rate, spectral features, harmonics, MFCCs) for 30-second tracks and 3-second windows, inspired by the GTZAN dataset, to increase data volume for training.

4.3 Model

The model used for generating pleasant jazz music is a Variational Autoencoder (VAE), comprising two main components: an encoder and a decoder. The encoder maps the input into a latent space, ensuring it follows a standard normal distribution, while the decoder reconstructs the input from this representation.Two hybrid models, VAE and Transformer, differ in processing recurrent data, with VAE using LSTMs and Transformer using Transformer mechanisms.The encoder consists of two LSTM layers (256 units each), followed by dense layers outputting the mean and log variance of a latent space with 64 dimensions. A sampling layer generates a latent vector from these parameters, making the latent space continuous for new data generation. The decoder mirrors the encoder with two LSTM layers (256 units), a dense layer to upscale the latent vector, and a final dense layer with SoftMax activation for predicting notes.

There are two types of hybrid models:

1. VAE that uses LSTM

2. VAE that uses Transformer

Model	Total loss	Reconstruction loss	KL
VAE-LSTM	4.8046	4.8003	2.7044e-05
VAE-Transformer	4.7117	4.6439	0.0677

Figure 4.2: Comparing LSTM and Transformer

4.3.1 VAE that uses LSTM

For music generation, a shorter input sequence is fed into the trained model, predicting notes iteratively. The input shifts by adding the predicted note and removing the first note until a sequence of 100 notes is generated. The result is converted into a MIDI file for playback, forming an AI-composed jazz song [5].

Training is time-intensive due to the high-dimensionality of musical data and the complexity of the model, which must effectively encode and decode representations.After training, the model generates music by taking a short input sequence, predicting a note, and iteratively shifting the input by adding the predicted note and removing the first note. This process repeats for 100 iterations, resulting in a sequence of 100 notes. The sequence is converted into a MIDI file to produce an AI-composed jazz song.Metrics tracked during training include reconstruction loss, Kullback-Leibler (KL) divergence loss, and total loss, calculated manually due to VAE's custom nature. Training is computationally intensive due to the high-dimensional musical data and model complexity.

The metrics tracked during the training process are the reconstruction loss, Kullback-Leibler (KL) loss, and the total loss, which are computed manually as Keras does not yet support the VAE pre-computed model. Training a Variational Autoencoder (VAE) can take considerable time due to several factors: musical data is often high-dimensional and complex, the model itself is intricate, incorporating LSTM and Dense layers, and the process requires learning to both encode and decode the representations accurately. After training, the model can make predictions by selecting a random shorter sequence from the training input, feeding it into the model to receive a predicted note.This process produces a sequence of 100 generated notes, which can then be transformed into a MIDI file and played as a new, AI-composed song.

- **Original Data** ($x_1, x_2, x_3, \ldots, x_n$): Represents the input sequence of data.
- **Encoder**:
  - **Two LSTM Layers (256 Units Each)**: Process the sequential data to extract temporal dependencies.
  - **Dense Layers** (64): Reduce the dimensions to generate the mean and log variance for the latent space.
  - **Latent Variable** (z): Encoded representation of the input, sampled from the Gaussian distribution defined by the encoder.
- **Latent Space** (z): Acts as the compressed representation of the input data. This representation is continuous and enables the generation of new data points.
- **Decoder**:
  - **Dense Layer** (64): Upscales the latent vector to match the input dimensions for decoding.
  - **Two LSTM Layers (256 Units Each)**: Decode the latent representation back into sequential data.
  - **Dense Layer** (*vocabulary*): Outputs a probability distribution over all possible chords or notes (via SoftMax activation).
- **Generated Data** ($x'_1, x'_2, x'_3, \ldots, x'_n$): Represents the output sequence generated by the model, aiming to resemble the original input data.
- **Loss Function**: Combines the reconstruction loss (difference between the original and generated data) and Kullback-Leibler (KL) divergence loss to optimize the model during training.

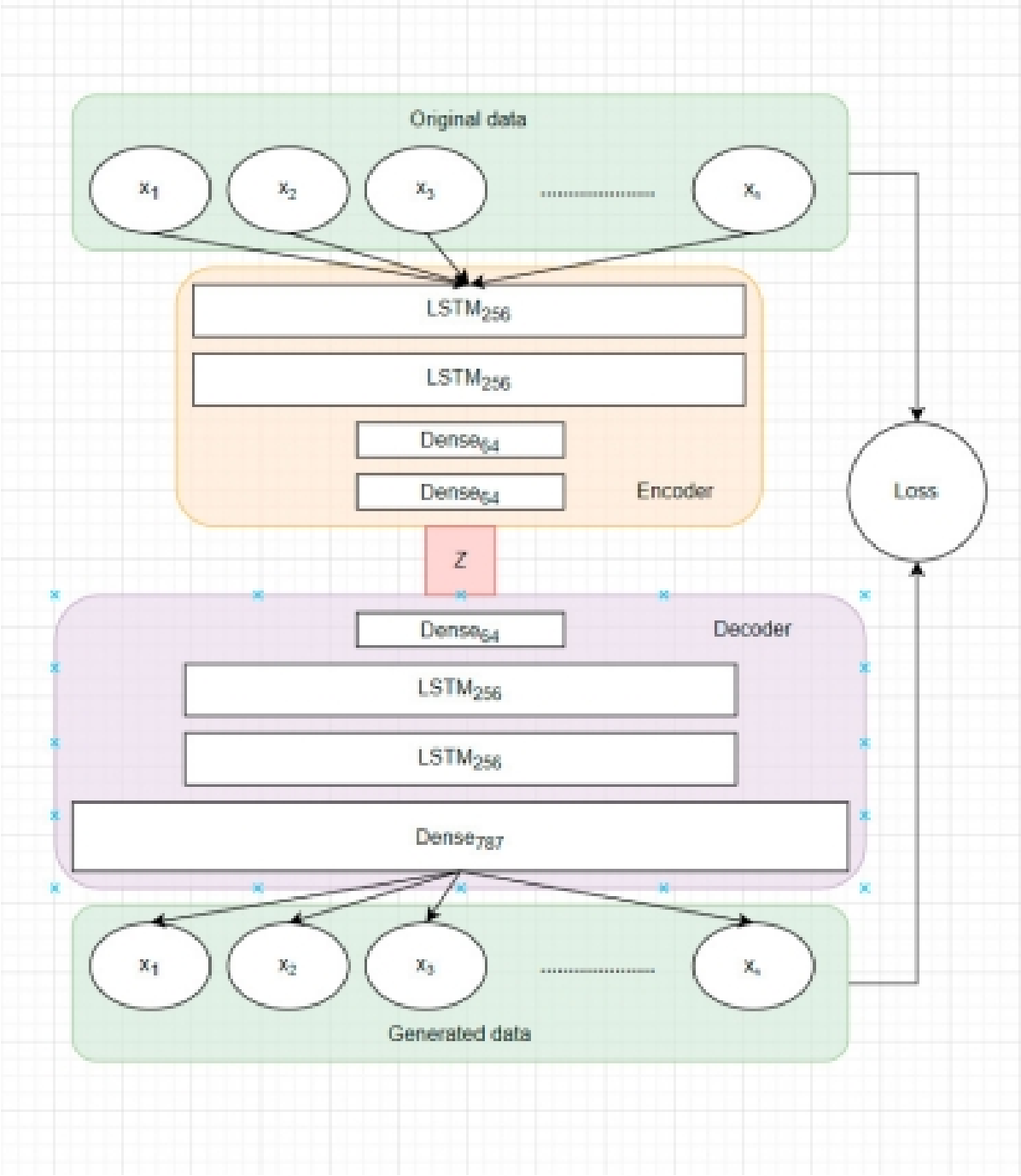


Fig. 2. Architecture of VAE that uses LSTM to encode/decode data.

Figure 4.3: VAE using LSTM

4.3.2 VAE that uses Transformer:

Transformers can be combined with Variational Autoencoders (VAEs) to leverage their attention mechanism, which effectively captures long-term dependencies in sequence data, such as rhythm and harmony in music generation. Unlike LSTMs, Transformers outperform in modeling long sequences and ensuring higher-quality outputs [5].

The model architecture replaces LSTM cells in the encoder and decoder with Transformer blocks. The encoder follows the standard Transformer design, including multi-head self-attention, position-wise feed-forward networks, Layer Normalization, and Dropout for regularization. The decoder maps points from the latent space through Transformer encoding and outputs reconstructed notes via a TimeDistributed layer.

1. **Original Data**: The input sequence, represented as $x_1, x_2, \ldots, x_n$.

2. **Positional Encoding**: A mechanism that adds position-based information to the input data, enabling the Transformer to understand the order of elements in the sequence.

3. **Transformer Architecture**:

    - **Attention**: A mechanism that focuses on the relevant parts of the input sequence, improving the model's ability to capture dependencies.
    - **Norm**: Layer normalization, which stabilizes and accelerates the training process by normalizing intermediate activations.
    - **Feed Forward**: Fully connected layers applied after the attention mechanism to process the data further.

4. **Dense Layers**:

    - **Dense64**: A dense layer that projects the encoded representation into a latent space of 64 dimensions, denoted as Z.
    - **Dense787**: A dense layer that projects the latent space representation back to the dimensionality of the generated data.

5. **Generated Data**: The output sequence, reconstructed by the model, represented as $x_1, x_2, \ldots, x_n$.

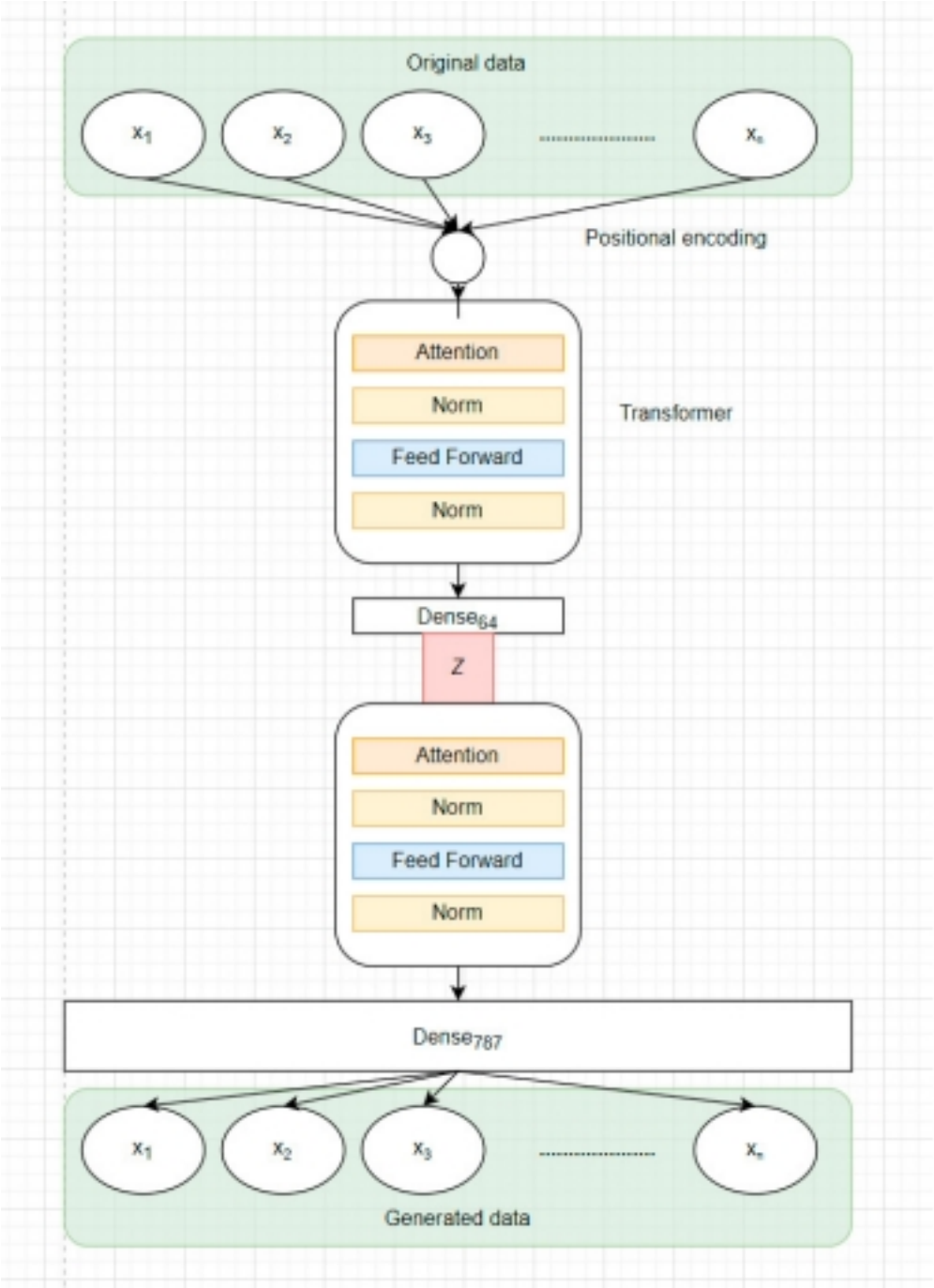


Figure 4.4: VAE using Transformer

The hybrid models, VAE-LSTM and VAE-Transformer, combine the strengths of VAE's with LSTMs and Transformers. VAE-LSTM integrates VAEs' rich latent learning with LSTMs' sequence modeling to generate coherent and nuanced music. VAE-Transformer utilizes attention mechanisms to capture long-range dependencies and intricate patterns, outperforming traditional sequence models in efficiency..

5. Conclusion

The presented work on music generation using AI-driven models seeks to emulate the creative process of human composers while incorporating an element of creativity and artistic expression. Through a detailed analysis of various methodologies and results, the research contributes to the understanding of how AI models can be refined for generating music that is both emotionally resonant and artistically compelling.

The future scope of this research includes the pursuit of more independent, creative music generation methods, the development of specialized architectures, and the integration of innovative concepts such as one-shot transfer. Additionally, advancements in evaluation metrics and computational resources will aid in refining the models and fostering more emotionally rich compositions.

Ultimately, the goal is to advance the state of AI-driven music generation not only from a technical standpoint but also by infusing the generated compositions with the complexity, depth, and emotional nuances that are hallmarks of human creativity.

6. Future Scope

In future research, there are several promising avenues to refine the task of music generation to more closely resemble human-produced music, while evoking deeper artistic sensibilities [5]. Key aspects to explore include:

- **Tailored Architectures for Musical Attributes:** Developing bespoke architectures designed for specific musical attributes may lead to the creation of compositions with enhanced emotional resonance and thematic coherence. This would further bridge the gap between human and AI-created music in terms of depth and emotional connection.

- **One-Shot Transfer:** Another intriguing avenue is the concept of "one-shot transfer," where both the stylistic essence and the content of music are distilled from a single audio track. This could potentially expand the creative possibilities, allowing the model to generate music that is more versatile and expressive.

- **Diverse Evaluation Metrics:** To improve the evaluation of the quality of generated music, it is crucial to incorporate a diverse array of mathematical and statistical measures. This would provide a more comprehensive framework for assessing the generated music, offering deeper insights into its effectiveness and quality.

- **Improved Hardware for Training:** Leveraging advanced hardware resources would allow for more extensive training epochs, which, in turn, could drive further improvements in the model's performance.

This multifaceted approach aims to advance not only the technical capabilities of AI-driven music generation but also to infuse the generated music with the richness of human emotion and artistic expression.

References

[1] J. D. Parker et al., “STEMGEN: A Music Generation Model That Listens,” ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Korea, Republic of 2024

[2] Emanuele Cosenza, Andrea Valenti, Davide Bacciu(2023). “Graph-based Polyphonic Multitrack Music Generation”. University of Pisa

[3] Peike Patrick Li, Boyu Chen, Yao Yao,Yikai Wang, Allen Wang, Alex Wang(2024),“JEN-1: Text-Guided Universal Music Generation with Omnidirectional Diffusion Models”. 2024 IEEE Conference on Artificial Intelligence (CAI)

[4] S Pavlova, Svetlana. (2023). Progressive distillation diffusion for raw music generation. 10.48550/arXiv.2307.10994.

[5] Tudor-Constantin, Pricop, and Adrian, Iftene (2024) ”Enhancing Music Genre Classification with Artificial Intelligence”, In 16th International KES Conference on Intelligent Decision Technologies (IDT-24), Santa Cruz, Madeira, Portugal, 19-21.June.2024.

[6] Shih-Lun, Wu, and Yi-Hsuan, Yang (2023) ”MuseMorphose: Full-Song and Fine-Grained Piano Music Style Transfer With One Transformer VAE,” in IEEE/ACM Transactions on Audio, Speech, and Language Processing, 31, 1953–1967, doi: 10.1109/TASLP.2023.3270726.

[7] Lu, Gening (2023) “Deep Learning-Based Music Generation”, Applied and Computational Engineering, 8, 366–379.

[8] Shih Yi Jen,Wu,ShihLun,Zalkow,Frank,Muller,Meinard,and Yang,Yi-Hsuan(2022) ”ThemeTransformer:SymbolicMusicGeneration with Theme-Conditioned Transformer”,2022.

[9] Defferrard, Michael, Benzi, Kirell, Vandergheynst, Pierre, and Bresson, Xavier (2017) “FMA: ADataset for Music Analysis”,

[10] GTZAN Dataset- Music Genre Classification. https://www.kaggle.com/datasets/andradaolteanu/ gtzan-dataset-music-genre-classification