Composing jazz music pieces using LSTM neural networks approach

Xincheng Zhang

School of Computer Science and Engineering, South China University of Technology, Guangzhou, 510641, China

1807050205@stu.hrbust.edu.cn

Abstract. The domain of artificial intelligence has increasingly extended into the creative arts, aiming to emulate and augment human creativity with automated processes. This is particularly evident in the field of music, where AI's ability to learn and produce intricate compositions has attracted significant attention. This study explores the challenge of generating jazz music using artificial intelligence, specifically focusing on the application of Long Short-Term Memory (LSTM) neural networks for jazz composition. By optimizing the model to address the genre's complexity, the research demonstrates the LSTM's capacity to capture and reproduce jazz's essential harmonic progressions and rhythmic nuances. Quantitative analyses show high accuracy and a deep understanding of musical structures, whereas qualitative feedback confirms the model's efficacy in producing compositions that embody jazz's spontaneity. Despite its achievements, the model's tendency to generate repetitive sequences suggests areas for improvement. This paper advances the field of AI in music, illustrating the potential of LSTM networks to mimic complex musical genres and emphasizing the necessity of ongoing model refinement to foster creativity. It highlights the evolving role of machine learning in music generation, proposing a foundation for future work aimed at diminishing the gap between AI capabilities and artistic expression.

Keywords: LSTM networks, jazz music, MIDI, model refinement.

1. Introduction

Computer music thus has developed and expanded with advancing technology, moving on from the rule-based systems and expanding wide with AI-based generative models. The four main approaches to music generation in the early nineties were large methods pertaining to a parameter-based model that required human-defined parameters or configurations and could be based on Markov chains, rule-based, or evolutionary algorithms. This shift toward the use of neural networks represented a great change: they started to produce better quality when generating music [1]. Further on, there were the parameter-based models, but at least they do not offer the same kind of rigidness, the ones that require specific input parameters and the ones that do not, like the prompt-based and visual-based models [1]. Noise2Music employs diffusion models to generate music clips from textual prompts, effectively capturing genre, tempo, instruments, mood, and era, illustrating a novel approach in music generation [2]. Despite its innovative application, the model exhibits a limitation with repetitive sequences and a limited vocabulary within songs, with the repetition displaying a consistent pattern, as indicated by cross-entropy analysis [3]. The MIDI technology made it possible for complex compositions through digital instrumentals. Besides, symbolic AI works with MIDI, dealing with harmony, melody, and structure, so it is very preferable for the tasks of harmonization and analysis [4].

Generative Adversarial Networks (GANs) have been utilized in models such as MuseGAN to generate multi-track music, effectively learning the distinct roles of different instruments within compositions [5]. This technique facilitates the production of harmonically coherent and stylistically consistent musical pieces. Additionally, models like Anticipation-RNN leverage autoregressive models to learn the conditional probability distribution of note sequences, thereby generating melodies that adhere to specific positional constraints [6]. The adoption of deep learning models, particularly Long Short-Term Memory (LSTM) networks, for music composition marks a significant advancement. Liang's approach incorporates LSTM-RNNs for automatic music synthesis, highlighting the use of audio as a medium [7]. Further research has demonstrated the capability of LSTMs to handle long-term dependencies effectively, underscoring their potential in music generation [8]. Central to the discussion on LSTMs is the computation of gradients, which elucidates how LSTMs process sequential data. The discussion also addresses the challenges of vanishing and exploding gradients, which are critical issues during the training of models on complex music compositions. Additionally, the transformer architecture has been effectively applied to various music generation styles, demonstrating significant success in managing the complexities involved in music production [9]. Reference [10] details a model designed to generate background music videos, employing two linear transformers to separately process note-related and rhythm-related music features, highlighting the significant relationship and independence between these features. Furthermore, Taiwan AI Labs introduced an innovative method utilizing dynamic directed hypergraphs for full-song music generation. This method represents musical events and complex compositions by considering multiple predictions at each time step, thus offering a sophisticated approach to music generation [11]. However, the music generation Transformer architecture seems to play toward what is capable in the future, not its existing flaws. Research undertaken so far suggests that there is an inherent weakness in Transformer models when it comes to tasks that require function composition or complex relationships between elements, and the composition involves an abstract, non-linear relationship between multiple elements [12]. While some models outperformed the Transformer models in some aspects, their sequential nature sometimes fails to capture the complex multidimensional structure in music, especially polyphonic compositions, where different sequences of notes arise simultaneously and not strictly in a linear order.

This research was provoked in an attempt to investigates the capabilities of Long Short-Term Memory models for processing, learning, and generation of jazz music generation, characterized by the complexity of melodic lines, harmonies, and polyrhythm. In the Methodology section, this will include an outline of the data collection, preprocessing process, design used in implementing the LSTM model, with supporting reasons for parameters used, and the specific Python libraries used. The next one is the experimental outcomes discussion, where the model performance is treated both quantitatively and qualitatively by comparing the generated pieces of music with the traditional one made in jazz. Finally, the conclusion summarizes in a nutshell the contributions of the study to the burgeoning field of AI-generated music, suggesting directions for future research to further refine and elaborate on this work.

2. Data and method

2.1. Data preparation

The dataset under investigation was derived from Doug McKenzie's extensive collection of high-quality MIDI recordings, encompassing both classical and jazz genres, available online at bushgrafts.com/midi. Initially consisting of approximately 6,855,000 pieces, the dataset underwent a filtration process to exclusively include monophonic sequences, utilizing a monophonic(stream) function, which reduced the dataset to 363,000 samples. This refinement was aimed at simplifying the dataset to enhance its suitability for model training purposes. Each piece within this dataset, represented in MIDI format, allows for the detailed analysis and manipulation of essential musical elements such as chords, durations, and tempos.

An additional function, extract_bpm_from_stream (stream), was developed to determine the beats per minute (BPM) for each piece, setting a default of 105 BPM in cases where the tempo was not explicitly indicated. The inclusion of BPM data aimed to introduce tempo variations characteristic of specific genres. The methodology proceeded with the iterative processing of MIDI files, employing the music21.converter.parse (file_path) for data loading and parsing, thereby extracting BPM, chords, durations, and keys. Subsequent phases involved the normalization of BPM values and the conversion of unique chords and durations into integer values, serving as a preparatory step to enhance the neural network's processing capabilities. The inversion of chord and duration dictionaries facilitated the reconversion of numerical representations back into their original musical notations. Training sequences were then constructed by analyzing songs to teach the model the prediction of future chords and durations, with sequence lengths designed to reflect temporal dependencies accurately. The conversion of data into NumPy arrays was carried out to ensure compatibility with TensorFlow/Keras frameworks, emphasizing the importance of BPM structure and the categorical encoding of chords and durations in augmenting the model's ability to interpret and learn from the dataset, particularly through one-hot encoding of chord and duration indices for discrete data processing.

2.2. Model

The LSTM-based model for jazz music composition integrates several layers and components to accurately capture musical sequences' intricacies. It is configured with a training batch size of 64, embedding dimension of 64, and a training duration of 100 epochs. This configuration optimizes the model's ability to represent input features while maintaining computational efficiency and achieving the necessary contextual understanding, evidenced by a sequence length of 32.

Embedding layers for chords and durations convert sparse categorical data into a more manageable, lower-dimensional format, facilitating the model's comprehension of the intricate relationships between musical notes and their timing. The inclusion of Beats Per Minute (BPM) as an input feature introduces tempo variations into the model, highlighting tempo's importance in crafting dynamic and expressive musical pieces, characteristic of jazz. The model's capacity for learning is centered around its LSTM layer, crucial for recognizing temporal dependencies and predicting patterns within music sequences. This illustrates a sophisticated grasp of music's sequential nature and the importance of context in generating cohesive musical pieces. To prevent overfitting, dropout layers are applied after the embedding and LSTM layers, promoting the development of more generalized and durable features by randomly excluding some layer outputs during training.

The output mechanism, consisting of dense layers with softmax activation functions, positions the model as a generative tool. This setup enables it to produce probabilistic predictions for upcoming chords and their durations, transforming it into an efficient means for generating music sequences. The choice of categorical crossentropy as the loss function for the multi-class classification task aligns with the model's objective to accurately predict the next chord and duration from multiple possibilities. The inclusion of a ModelCheckpoint callback underscores the strategic retention of the model's best-performing iteration based on loss metrics, automating the preservation of peak performance without manual oversight.

2.3. Evaluations

The performance of the music generation model was assessed using evaluation metrics and methodologies. Accuracy and loss metrics, recorded throughout the training process, were compared against a reserved test dataset to measure the model's effectiveness. Graphical representations of accuracy and loss for training and validation datasets across epochs were plotted to observe the model's learning progression and identify potential overfitting occurrences.

A nuanced approach to learning rate adjustment is implemented through the custom learning rate scheduler, i.e., the WarmUpCosineDecayScheduler. This scheduler marries a warm-up phase, where the learning rate incrementally reaches a base level, with a cosine decay phase for its gradual reduction. Such adjustment is instrumental in enhancing training stability and performance, catering to both the acceleration of initial learning and the precision of subsequent tuning phases. This adaptability ensures the scheduler's applicability across various training scenarios and dataset scales. Utilizing the Adam optimizer in conjunction with this custom learning rate schedule harnesses Adam's adaptive advantage while customizing learning rate adjustments for optimal training outcomes.

Utilizing the music21 toolkit, analyses of harmonic progressions, melodic contours, and rhythmic patterns were performed, employing metrics such as note density, pitch range, and motif repetition to quantitatively discern the musical pieces' characteristics. This analytical approach facilitated assessments of the generated music's internal consistency and coherence, ensuring the compositions adhered to recognizable musical standards. Furthermore, a regimen of comprehensive experimentation involving the adjustment of LSTM layers, embedding dimensions, and input features was undertaken to enhance model performance.

3. Results and discussion

3.1. Model Performance

The initial approach involved the analysis of 49 MIDI segments, each comprising 16 measures, to evaluate the model's efficacy, designating 30% of the training dataset for validation purposes. As depicted in Figure 1, the validation accuracy exhibits high levels from the onset, suggesting effective generalization. However, post-epoch 50, a noticeable decline in validation accuracy, particularly in chord output accuracy, indicates potential overfitting or difficulties in generalizing to the validation set. Concurrently, validation loss, initially minimal, begins to rise significantly after approximately epoch 50. This divergence, characterized by increasing validation loss amidst stable or decreasing training loss, is indicative of overfitting, wherein the model's performance improves on the training dataset but deteriorates on the validation dataset.

The learning rate strategy's positive impact implies the utility of incorporating a warm-up phase followed by cosine annealing. The initial warm-up phase permits a gradual commencement of learning, averting premature extensive updates. Subsequent cosine annealing fine-tunes the learning rate for smooth convergence to a minimum, aiding in the avoidance of local minima that could obstruct learning. These strategies collectively facilitate a more refined and efficient model training regimen, as evidenced by the observed improvements in model performance as shown in Figure 2.

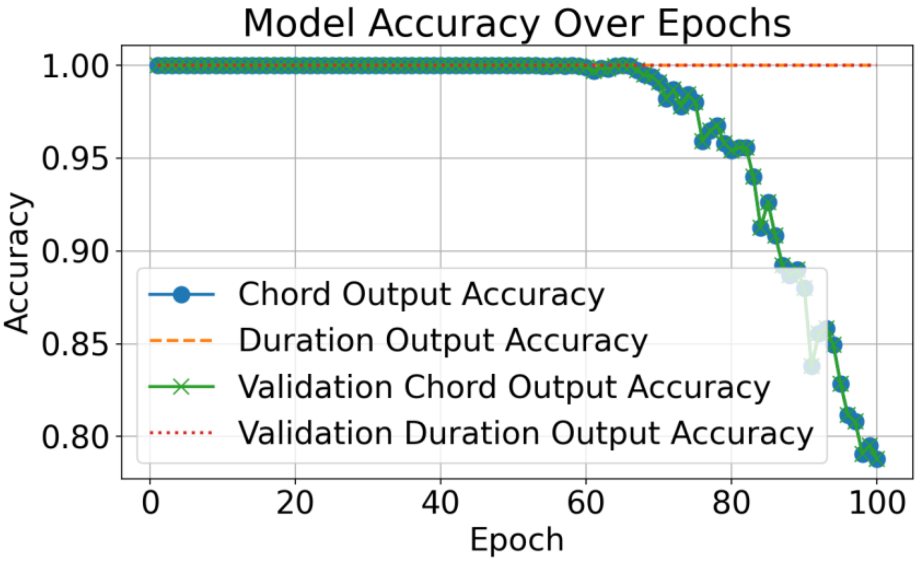

Figure 1. Model Accuracy over Epochs (Photo/Picture credit: Original).

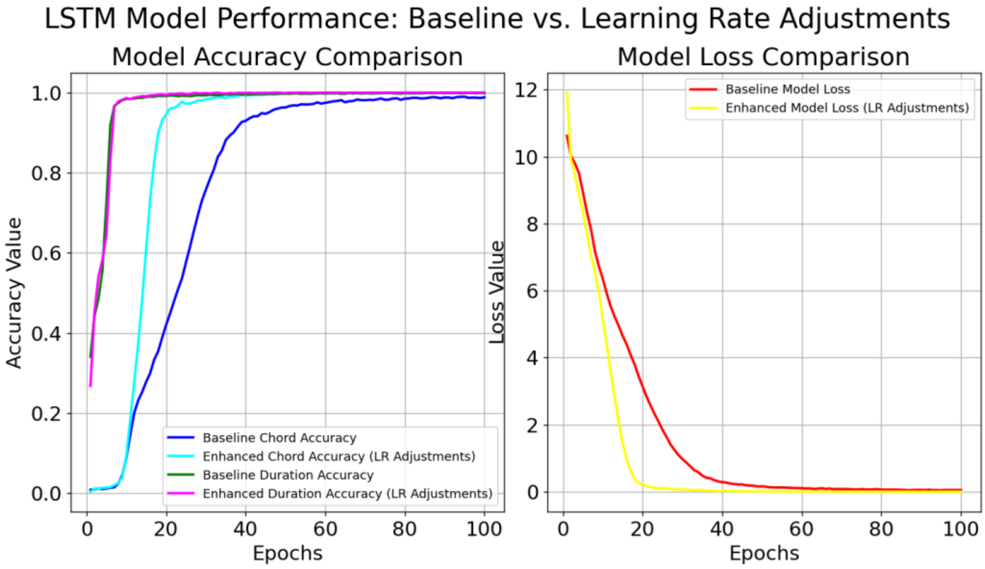

Figure 2. LSTM Model Performance: Baseline vs. Learning Rate Adjustments (Photo/Picture credit: Original).

Table 1. LSTM model structure hierarchy table.

Layer(type)	Output Shape	Param #	Connected to
binary_input (InputLayer)	(None, 128)	0	---
chord_input(InputLayer)	(None,128)	0	---
duration_input(InputLayer)	(None,128)	0	---
reshape (Reshape)	(None,128,1)	0	binary_input[0][0]
embedding(Embedding)	(None,128,64)	684992	chord_input[0][0]
embedding_1(Embedding)	(None,128,64)	4032	duration_input[0][0]
bpm_input(InputLayer)	(None,128,1)	0	---
conv1d (Conv1D)	(None,128,64)	128	reshape[0][0]
concatenate(Concatenate)	(None,128,192)	0	embedding[0][0];embedding_1[0][0]; bpm_input[0][0]; conv1d[0][0]
dropout(Dropout)	(None,128,192)	0	concatenate[0][0]
LSTM(LSTM)	(None,512)	1443840	dropout[0][0]
dropout_1(Dropout)	(None,512)	0	LSTM[0][0]
dense(Dense)	(None,256)	131328	dropout_1[0][0]
chord_output(Dense)	(None,10703)	2750671	dense[0][0]
duration_output(Dense)	(None,63)	16191	dense[0][0]

The revised model exhibited a marginal improvement in chord output accuracy relative to the baseline model, suggesting that the implemented learning rate strategy enhanced the model's capability to discern the dataset's intricacies. This enhancement is attributed to a more effective exploration of the weight space during training, which likely contributed to optimizing the learning process and achieving increased accuracy. Furthermore, the stability and high performance in duration output accuracy were maintained across both the baseline and adjusted models. This consistency indicates that the modifications to the learning rate did not detrimentally impact the model areas already performing adequately. It underscores the effectiveness of the learning rate strategy in preserving the model's strengths while facilitating improvements in other areas. A notable reduction in loss was observed in the adjusted model across epochs, indicating a more efficient error minimization between predicted and actual values. This study expanded the dataset to encompass MIDI files representing complete tracks, thereby introducing longer durations and increased variability. Adjustments to the sequence length were made, ultimately setting it at 128, to observe the impact on accuracy and loss metrics. The incorporation of tempo, alongside chords and duration, was hypothesized to result in audio clips of enhanced variation. Further, the inclusion of note onsets and rests aimed at achieving nuanced rhythmic changes, as illustrated in Table 1. Subsequent analysis of training and validation accuracy and loss across epochs, as depicted in Figure 3, delineated the model's learning trajectory, highlighting the iterative refinement of model inputs to enhance musical output generation.

The model initially exhibits low chord accuracy, beginning at approximately 0.65% at epoch 1, indicating challenges in accurately predicting chord sequences. Chord accuracy significantly improves as training progresses, reaching 24.78% by epoch 24 and 83.53% by epoch 62. By epoch 100, accuracy peaks at 99.67%, demonstrating marked proficiency in chord prediction. Duration accuracy starts higher, at about 41.73%, reflecting a better initial understanding of duration patterns. It increases to 58.38% by epoch 12 and continues rising, reaching 90.87% by epoch 100, indicating a strong grasp of duration patterns. Training loss starts at 5.8722 and steadily declines, signaling effective learning, and reaches 3.3721 by epoch 100. These trends highlight the model's progress in learning and optimizing its predictive capabilities throughout the training.

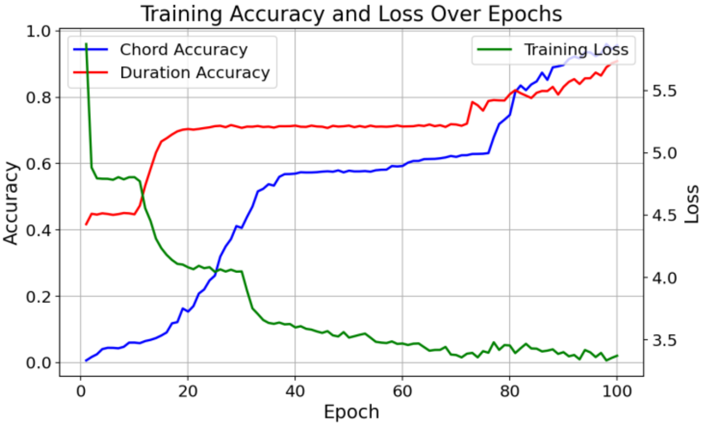

Figure 3. Training Accuracy and Loss Over Epochs (Photo/Picture credit: Original).

3.2. *Music Generation*

Figure. 4 shows the extracted of the initially generated music from the model experiment displayed in music21. The segment shows a variety of chords, including G3 D4 B-4 D5 G5 (suggests a G dominant 7th chord with extensions, which is common in jazz); A3 C4 F#4 A4 (an A major 7th chord); G4 B4 C#5 F#5 (a G major 7th chord); G#3 D4 G4 B-4 F5 B-5 (a G#7th chord); E2 G3 B-5 (could be part of a larger harmonic movement typical in jazz). The rhythmic patterns here show some variation in duration and placement, with chords occurring at off-beat timings such as 0.75, 1.25, 3.25, which suggests a level of rhythmic complexity.

After the inclusion of rhythmic parameters such as tempo and onset into the input, the music segment generated by the model is depicted in Figure 5. F3 G#3 C#4 could be interpreted as an F minor chord with added tones, indicating jazz's harmonic richness. The chords F3 C4 F4 A4 C5 suggest F major 7th chords, another hallmark of jazz harmony. Chords like C#3 G#3 C#4 B-4 could be seen as altered or extended chords, which are prevalent in jazz for creating tension and release. Jazz often features syncopation, swung rhythms, and a greater emphasis on beat subdivision, contributing to its distinctive "groove." This segment's rhythmic pattern, with chords placed on and off the beat (e.g., at 0.25, 0.5, 1.0, 1.5), suggests a level of rhythmic complexity that could align with jazz.

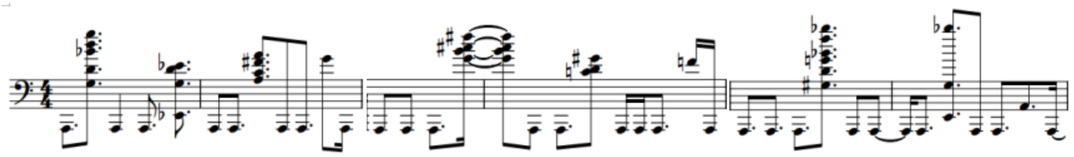

Figure 4. Three Measures Extracted of the Initially Generated Music (Photo/Picture credit: Original).

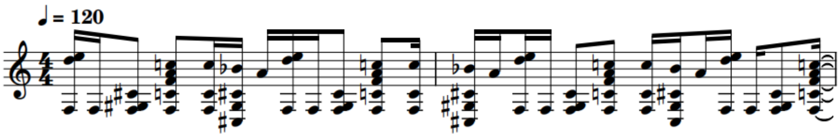

Figure 5. Generated Music Segment from the Improved Experiment (Photo/Picture credit: Original).

3.3. Interpretation of results

Jazz music is characterized by its extensive use of complex harmonic structures, such as seventh, ninth, eleventh, and thirteenth chords, alongside various chord substitutions and alterations. It also features significant rhythmic complexity, incorporating elements like syncopation, swing rhythms, and an improvisatory feel. Subsequent to the dataset's final expansion and the completion of training, the music segment produced by the model is illustrated in Figure 6.

The model demonstrates significant potential in generating music that aligns with both the theoretical and stylistic dimensions of jazz, including a profound capacity to learn and reproduce complex jazz patterns such as extended chords, chromatic movements, and dissonances—elements quintessential to jazz music. Chord sequences such as E-2 G3 B-3 D4 F4 and F#3 C#4 F#4 B-4 B4 E-5 reveal intricate chord structures characterized by alterations and extensions. The occurrence of minor second intervals (evidenced by structures like B-5 C6) and tritone substitutions (indicated by sudden changes in root motion) are consistent with jazz's inclination towards creating harmonic tension followed by resolution. Furthermore, the timing of chord transitions suggests the employment of syncopation and off-beat accents, rhythmic features synonymous with jazz. This transition between complex, dissonant chords and simpler, consonant voicings reflects a dynamic exploration of harmony that mirrors jazz's thematic focus on tension and release.

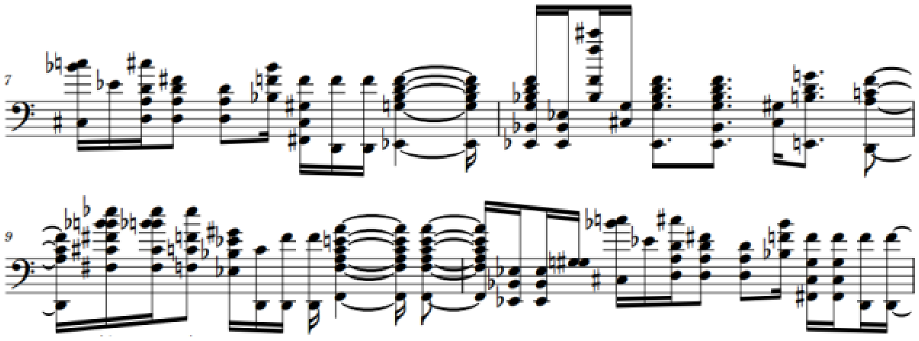

Figure 6. Generated Music Segment from the Final Training (Photo/Picture credit: Original).

4. Limitations and prospects

The model exhibits a notable capability in encapsulating jazz music's core elements but tends towards repeating distinct patterns. This repetition may originate from an excessive dependence on sequences that are prevalent within the training corpus, potentially resulting in compositions of extended length that lack innovation. Moreover, although the model efficiently navigates numerous aspects of jazz theory, it occasionally produces progressions or note selections that stray from conventional jazz practices. Such deviations could compromise the musical coherence anticipated by experienced listeners, suggesting a requirement for more sophisticated training methodologies or subsequent theory-informed modifications.

A further limitation lies in the model's partial representation of the dynamic and articulative expressions intrinsic to live jazz performances. This shortfall might yield compositions that, despite their harmonic and rhythmic complexity, do not fully convey the expressive depth and emotional resonance characteristic of human-rendered music. Additionally, certain compositions from the model may exhibit structural repetitiveness and predictability, detracting from the spontaneity and improvisational essence highly valued in jazz music. To address these limitations, future research directions could include the adoption of variational autoencoders (VAEs) or the incorporation of randomness into the generative process to enhance compositional diversity and unpredictability. Direct integration of music theory into the training methodology, through rule-based systems or tailored loss functions that deter theory discrepancies, could ensure greater adherence to musical principles.

To further enhance pattern diversity and foster creativity, subsequent enhancements to the model could include the development of mechanisms specifically designed to minimize sequence repetition, alongside the inclusion of a more varied training dataset. Moreover, to rectify issues related to musical theory compliance, the integration of in-depth music theory knowledge into the model's training regimen could prove beneficial. This integration could be operationalized through the application of rule-based systems or the establishment of training objectives that are informed by music theory principles. Further adherence to jazz conventions could be achieved through the imposition of theory-based constraints or the application of post-generation refinements rooted in jazz theory, thereby improving the musical coherence of the generated compositions. Significant advancements in the model's capability could also be realized by explicitly incorporating elements of dynamics and articulation within the training paradigm. This could entail treating these elements as separate dimensions within the training data or devising model components expressly for their articulation. Lastly, the investigation of neural network architectures that are better suited to grasping the long-term structural and improvisational nuances of jazz could lead to the generation of compositions that are both more engaging and unpredictable. Such architectural explorations promise to address the current model's limitations while paving the way for generative music systems capable of producing outputs that more faithfully represent the depth and diversity of jazz music.

5. Conclusion

To sum up, this study embarked on an exploration of automated jazz music composition utilizing a Long Short-Term Memory (LSTM)-based model, revealing significant insights into the capabilities and challenges of current AI-driven music generation. Through a detailed configuration involving a specific training batch size, embedding dimension, and epoch count, the model demonstrated an optimized ability for input feature representation, computational efficiency, and contextual understanding. The effectiveness of the model was rigorously assessed using accuracy and loss metrics, compared against a test dataset, and further analyzed through the implementation of a nuanced learning rate adjustment strategy, the WarmUpCosineDecayScheduler. Utilization of the music21 toolkit facilitated in-depth analysis of harmonic progressions, melodic contours, and rhythmic patterns, underscoring the model’s proficiency in learning, and reproducing complex jazz patterns, including extended chords and chromatic movements. However, the analysis also uncovered a tendency for pattern repetition within the generated compositions, suggesting a potential area for further refinement. The model's proficiency in capturing the essence of jazz, despite some limitations, underscores the feasibility of employing LSTM neural networks in the field of automated music composition. It opens avenues for future research aimed at enhancing the diversity and creativity of AI-generated musical pieces.

References

[1] Zhu Y, Baca J, Rekabdar B, et al. 2023 A Survey of AI Music Generation Tools and Models arXiv preprint arXiv: 2308.12982.

[2] Huang Q, Park D S, Wang T, et al. 2023 Noise2music: Text-conditioned music generation with diffusion models arXiv preprint arXiv: 2302.03917.

[3] Dai S, Yu H and Dannenberg R B. 2022 What is missing in deep music generation? a study of repetition and structure in popular music arXiv preprint arXiv: 2209.00182.

[4] Briot J P, Hadjeres G, Pachet F D 2017 Deep learning techniques for music generation--a survey arXiv preprint arXiv: 1709.01620.

[5] Ji S, Luo J, Yang X 2020 A comprehensive survey on deep music generation: Multi-level representations, algorithms, evaluations, and future directions arXiv preprint arXiv:2011.06801.

[6] Briot J P and Pachet F 2017 Music generation by deep learning-challenges and directions arXiv preprint arXiv:1712.04371.

[7] Liang M 2022 Mobile Information Systems vol 2022 p 7618045.

[8] Ghojogh B and Ghodsi A 2023 Recurrent neural networks and long short-term memory networks: Tutorial and survey arXiv preprint arXiv: 2304.11461.

[9] Huang Y S and Yang Y H 2020 Pop music transformer: Generating music with rhythm and harmony arXiv preprint arXiv: 2002.00212 3.

[10] Yang X, Yu Y, Wu X 2022 Applied Sciences vol 12(10) p 5050.

[11] Hsiao W Y, Liu J Y, Yeh Y C, et al. 2021 Proceedings of the AAAI Conference on Artificial Intelligence vol 35(1) pp 178-186.

[12] Peng B, Narayanan S and Papadimitriou C 2024 On Limitations of the Transformer Architecture arXiv preprint arXiv:2402.08164.