Gamelan Melody Generation Using LSTM Networks Controlled by Composition Meter Rules and Special Notes

Arry M. Syarif[1,*], A. Azhari[2], S. Suprapto[2], and K. Hastuti[1]
[1] Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang, Indonesia; Email: afis@dsn.dinus.ac.id
[2] Universitas Gadjah Mada, Indonesia; Email: {arisn, sprapto}@ugm.ac.id
*Correspondence: arry.maulana@dsn.dinus.ac.id

***Abstract*—This study proposes a Gamelan melody generation system based on three characteristics, which are the melodic patterns recognition, composition meter rules that control the duration of notes, and the special notes (pitches) selection which represent ambiguous rules in determining the Gamelan musical mode system. Long-Short Term Memory (LSTM) networks were trained using the sequence prediction technique to generate symbolic based Gamelan melodies. The dataset collected from sheet music was converted into ABC notation format, added with codes representing the composition meter and special notes, and restructured into a character-based representation format. The LSTM network training showed good results in the melodic patterns recognition but the networks take less than 10 attempts for the LSTM network to successfully generate one melody. The evaluation was conducted using experts' judgment. Three generated melodies were sent to experts to be read, hummed and judged. Overall, the evaluation results showed that the generated melodies can comply with the characteristics of the Gamelan melodic patterns, the composition meter and the special notes.**

***Keywords*—melody generation, melodic patterns, symbolic music, sequence prediction, long-short term memory**

I. Introduction

This research aims to develop a melody generation system for Javanese traditional compositions known as Gamelan music. The idea was to develop a system to learn all the musical elements contained in sheet music, then generate melodies based on the learning results. The characteristics of Gamelan music require a different approach from Western music or other musical genres. Gamelan experts adhere to the *pakem* (Gamelan melody rules), or standardization that requires compositions not deviating from the melodic patterns inherent in existing compositions from the past. However, although the *pakem*, the narrow sense of the word, can be interpreted as rules, there are no rules that are clearly and rigidly formulated. For example, say composition X and Z have different styles, in order to maintain the distinction between the two compositional styles, certain notes can be used at the end of the bars in composition X but not in composition Z. Moreover, such forbidden notes can be used at the end of the bars in composition Z by selecting the appropriate notes in a certain order so that the prohibition can be lifted. When asked to elaborated upon, experts can provide examples but they cannot provide clear rules since the notes that can lift the prohibition may be located anywhere based on different contexts. The same result was obtained when extracting deeper information related to context but there are yet more examples of contexts in which no clear and rigid rules apply. There are still many more ambiguities in understanding the rules of Gamelan music. However, it is an interesting fact considering the characteristics of the Gamelan melody that can still be maintained today. For a Gamelan expert (or enthusiast), the melody that is played or hummed can be easily recognized as being a Gamelan composition or not. The details of the complex and sophisticated characteristics of Gamelan music are not discussed in this article, since the melody generation approach proposed in this study does not seek to define the rules of Gamelan music that are broad and ambiguous but rather study its melodic patterns controlled by the composition meter rules and special notes.

Melody has a complex time series data structure, and the LSTM method is suitable for solving problems in melody generation as worked by [1-5]. Hence, the LSTM method is used to generate Gamelan melodies based on the melodic patterns controlled by the composition meter rules and special notes. The composition meter can be measured quantitatively. It controls the notes duration and the number of notes in bars, lines, and the melody length based on the time signature. Meanwhile, the special notes that control the musical mode of melodies deals with the *pakem* i.e., rules that are not clearly defined and not quantitatively measurable.

In brief, the LSTM network is used to generated melodies by learning the melodic patterns controlled by the special notes. Furthermore, the generated melodies are declared as legit melodies if they can comply with the composition meter rules. In other words, the rules are used to evaluate the ability of the LSTM networks in recognizing the structure of beats, bars and line when generating melodies.

Manuscript received May 22, 2022; revised July 5, 2022; accepted July 22, 2022; published February 10, 2023.

doi: 10.12720/jait.14.1.26-38

II. Related Works

The melodic patterns and composition meter are easier to identify and learn through sheet music than to analyze them through audio data. Conversion of audio data into sheet music was conducted by [6] and the results achieved an accuracy of 78% when comparing the generated sheet music against the original version. In a different context, the generated sheet music is still lacking accuracy when used to learn and store musical knowledge. Compared to a source that derives from the data conversion process, original sheet music ensures the validity of its content. In such circumstances, sheet music written by musicians is considered the best source to get knowledge of compositions.

The LSTM network has been applied in various melody generation works and mapping of musical notes. Input and output are very important in melody generation modelling [7, 8], such as music generation works by [2-5] which used only the LSTM network to learn the composition structure and generate music. Meanwhile, some researchers combine the LSTM network and Convolutional Neural Network (CNN) to generate music based on the separation of tasks. A music generation system using the CNN to learn the composition structure and the LSTM network to generate melodies was developed by [9]. Meanwhile, a music generation system using the CNN to extract features and the LSTM network to generate melodies was developed by [1].

Traditional Western sheet music based on notes and the composition structure was encoded by [3], where the composition structure is measured using the bar length, beat level and beat position. On the other hand, Royal *et al.* [5] focused on the time segmentation based on the time signature to generate unique music by mixing different musical styles collected from American and British folk songs dataset. The data segmentation to learn chord inversion shapes in order to generate background music was used by [2]. The composition meter in generating music was also used by [4, 9]. Although the composition meter is one of musical knowledge that is important in representing music, some researchers deliberately do not use it, such as [10] who used the LSTM network to generate music based on the MusicXML format of Bach chorales, where only notes and rhythms were used for the representation, whereas the composition structure was not explicitly represented, and [11] also did not use the composition meter to be learned by the LSTM network. The success of not explicitly using the metrical structure as a feature and passing it on to the network through learning the note sequence patterns remains to be tested. There is a possibility that the type of music influences whether the metrical structure should be used as a feature or not. In other words, not explicitly using the metrical structure as a feature may be accomplished in certain types of music which contains melodic patterns that can clearly show the compositional structure. Moreover, whether a network can discern the time signature without explicitly learning the composition structure is questionable. However, in many studies, the metrical structure is one of the fundamental feature that machines must recognize in generating music. Therefore, it is needed to be used as a feature that can strengthen network performance in order to satisfy not only the melodic patterns but also the metrical structure.

Many works use MIDI data as the source, such as works by [1, 9, 11, 12]. The Mp3 to MIDI format conversion was done by [11] in order to set input of a sequence consisting of notes and chords. The smaller the number of features, the lighter the computational load to train the network. 36 of the 128 notes of MIDI data based on the notes found in the dataset was used by [9], while [13, 14] used 48 and 88 notes of MIDI data, respectively. Symbolic music generation was developed by [15] using the Generative Adversarial Network (GAN) framework and the CNN method, and the output is multi-track music generated bar by bar. The temporal structure model, which uses the previous bar to generate the next bar, was used to maintain coherence between bars. Meanwhile, [16] used a note-by-note generation technique using a rule-based method with sequential mining techniques to generate music bar by bar, and inter-bar coherence was carried out based on the last note on the previous bar to determine the first note in the next bar. Text-based format with characters-based or words-based representations techniques—which is part of the bag-of-words model, a model of a text representation in natural language processing that represents text as a token—is also commonly used to represent music. Data mapping is used to set a feature in the characters-based and words-based representation, where the musical elements are converted to strings, characters, or words and arranged to acquire unique characters or words selected as features. Bretan *et al.* [17] used a unit selection method, while Huang *et al.* [18] built a vocabulary based on the dataset to set a sequence of discrete token. The character-based representation was used by [4] to set a feature consisting of 87 unique characters out of a total 155,222 characters. Meanwhile, Choi *et al.* [19] used a word-based representation and set features consisting of 39 unique words out of a total of 539,609 words. The rate of success in using these musical representation techniques depends on the strategy data mapping while still paying attention to the efficiency of the number of features produced.

Musical elements information of bar, beat and notes was processed using three LSTM networks, where one LSTM network was to generate bars, and another LSTM network was to generate beats, and then generated bars and beats were used as the template for the last LSTM network to generate a note sequence [20]. In this research, instead of using more than one LSTM network, the time, beat and notation information that can be measured quantitatively based on the length of the duration of each of these elements can be represented in an input vector to be studied by a single LSTM network. Meanwhile, Yu *et al.* [21] used the LSTM-GAN to generate melody based on lyrics involving note, note duration and rest aligned with the syllables of the lyrics. Both the authors of [20] and [21] used information that was measured quantitatively, namely the composition meter.

Unfortunately, there is no description that explains that the developed system can always generate a melody in one process. The ability of the system to generate melodies that comply with the composition meter rules will be more reliable if the measurements are included. In this research, the design of the system was to include composition meter measurements to evaluate the output of melody generation, and to provide an opportunity for the LSTM network to repeat the melody generation process if there is a discrepancy between the duration of notes, beats, bars and lines against the composition meter.

III. Gamelan Music

A. The Musical Scale and Musical Mode Systems

Karawitan, also known as Gamelan music, is a traditional music ensemble from Java that uses Gamelan as the musical instruments and *gendhing* as its song. Gamelan consists of two musical scale systems (*laras*) with different tuning systems and audio frequencies, which are the *slendro* and the *pelog*. The *slendro* musical scale system consists of five notes, which are 1, 2, 3, 5, 6. Meanwhile, the *pelog* musical scale system consists of seven notes, which are 1, 2, 3, 4, 5, 6, 7. There is a musical mode system (*pathet*) that influences the sound characteristics of Gamelan compositions. The *slendro* musical scale consists of three musical mode systems; *manyura*, *nem* and *sanga*. The *pelog* musical scale also consists of three musical mode systems; *barang*, *lima* and *nem*. Although the musical mode system of *nem* exists in both musical scale systems, the sound characteristics of their compositions are different.

Based on the related literatures and discussions with Gamelan experts, it is difficult to explain and formulate how the music mode system works. To put it briefly, the musical mode system is controlled by dominant notes in a particular order, such as in the last note of certain bars or lines. According to [22], "When native musicians hear a *gendhing* (Javanese: song), they immediately recognize the *pathet*. It is, however, very difficult to explain how they know it (p. 34)". Meanwhile, Ishida [23] stated that '*pathet* is, in a very limited sense, a zone of notes arranged in a certain order. A certain *pathet* occupies a higher or lower zone than other *pathet* in the total range (p. 492)". Moreover, Becker [24] also emphasized that *pathet* is not simple; it gives meaning to compositions based on the melodic pattern. Learning the musical mode system is challenging, and a grey area in itself. There are no rules that explicitly or specifically define the musical mode system of Gamelan compositions.

B. Melody and the Melodic skeleton

Gamelan compositions consist of a series of *gatra* whose function is similar to the bars in Western music. Each *gatra*, hereinafter referred to as the bar, consists of four beats, and each beat contains one or more. The number of which is determined based on the time-signature and note duration. For example, in a composition with a time signature of 1/2, each beat contains one or more notes for a total duration of 2, where the note duration for each note can be 0.25, 0.5 or 1. In Gamelan compositions, one line consists of two bars, while the length of the composition is determined based on the various types of compositions, such as *lancaran, ladrang, ketawang, srepegan*, etc. For example, *lancaran*-type compositions consist of 16 beats equivalent to four bars and two lines, while *ladrang*-type compositions consist of a minimum of 32 beats equivalent to eight bars and four lines. In addition to the length of the melody, the different types of compositions are based on the purpose of the composition being played and the complexity of the rendition. For example, the type of *lancaran* with simple play functions as an independent musical repertoire or to accompany the entrance of the dancers or the battle in puppet performances, while the type of *ladrang* has a slightly more complicated level with functions as an independent musical repertoire as well as accompaniment to puppet or dance performances [25]. In this research, for no particular reason, the dataset used a collection of *ladrang-type* compositions.

Gamelan compositions contains melody and its melodic skeleton. A melodic skeleton called *balungan* (Javanese: skeleton) is an abstraction form of a melody, and it can be analogous to a chord in Western music. There are three elements that underlie the performance of Gamelan, i.e. melody, melodic skeleton, and type of composition. Gamelan instruments are grouped according to these three elements. The group of instruments that play melodic skeletal parts called *ricikan balungan* (melodic skeleton) includes *demung, saron, peking* and *slenthem*, and the group of instruments that play melody parts called *ricikan garap* includes *gender, rebab* and *suling*, while the group of instruments that play notes that determine the type of composition called structural *ricikan* includes *gong, kempul, kethuk* and *kempyang*.

C. The Composition Meter

Gamelan sheet music displays elements of melody and its melodic skeleton in different lines sequentially, while elements that determine the type of composition are displayed in the melodic skeleton part. Meanwhile, in the melody part, the dotted symbols above or below the notes, the horizontal lines symbols and curve lines below the notes are symbols that function as in Western music, which are the high-low notes, notes duration and legato marks. In addition, there are dotted notes both in the notes sequence of melody and melodic skeleton that represent moments of silence. The time signature in Gamelan music is divided by multiples of the values 1/1, 1/2, 1/4, 1/8 and so on. The numerator of the time signature refers to the note duration value for each beat in the melodic skeleton. Thus, beats in the melodic skeleton always have a value of 1, and based on sheet music data analysis, each beat contains one note so that each note in the beat always has a note duration value of 1. It can be concluded that the beat in melodic skeleton has a constant value of 1. The denominator of the time signature refers to the note duration value for each beat in the melody. Thus, each beat may contain one or more notes controlled by the denominator value in the time signature and the note duration value. Fig. 1 uses the first line of a composition played with the time signature of 1/2 to illustrate the note, beat, bar and line durations. In this illustration, the single and double horizontal lines above the notes indicate the note duration values of 0.5 and 0.25, respectively, whereas the unmarked notes have the note duration value of 1. In the illustration, each beat in the melodic skeleton consists of one note so the beat duration value is equal to the note duration value, i.e., 1, while each beat in the melody which has the beat duration value of 2 consists of notes with a total note duration value of 2.

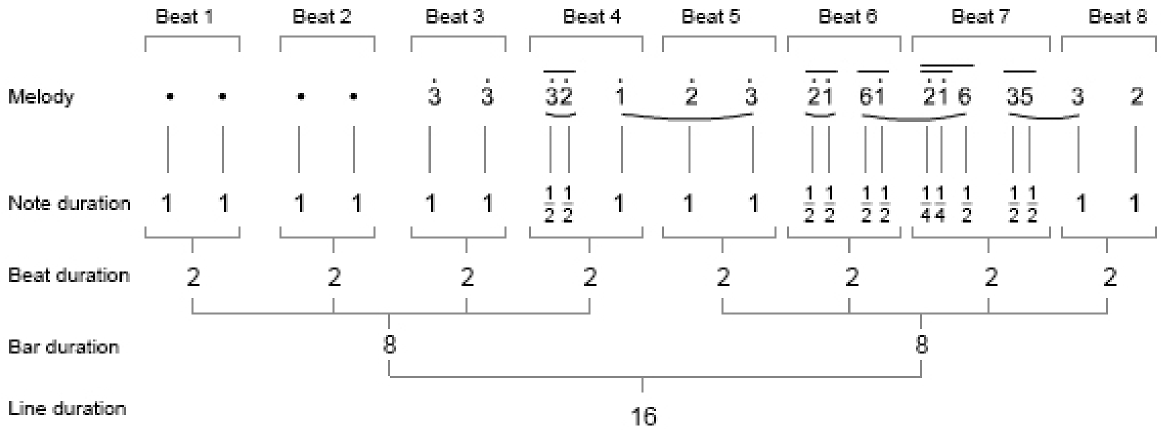


Figure 1. An illustration of duration of notes, beats, bars in a line for a composition played in the time signature of 1/2.

IV. Proposed Methods

This study was limited to melody generation, so melodic skeletal data were not used. The melody generation was performed by generating note sequences using the note-by-note generation technique which was then measured using the composition meter rule. A collection of Gamelan sheet music containing symbolic data used as the data source was converted into a text-based format so that it could be processed computationally. The *Gendhing* Scientific Pitch Notation (GSPN) model proposed by Syarif *et al.* [26] was used to convert Gamelan sheet music into a text-based format. The GSPN model which is a text-based notation writing system for Gamelan music contains note sequence and legato sign information. Data in GSPN format have no information about beat, bar and line. Notes that function to control the musical mode system or special notes are not represented either. Therefore, the addition of the code was needed to complete the information not yet represented in the GSPN model.

Composition meter which contains information of note and beat was added into the melody data using coma and space symbols, respectively. Meanwhile, composition meter which contains bar, line and melody length information was added into the melody data using special characters. The special notes that control the musical mode system were determined based on the last note in the bars and lines. Based on the literatures and interview with experts, the notes in these positions influence the musical mode system of compositions. Information of these notes was added into the data melody using special characters as well. This was done by producing the composition meter rules which were then developed into algorithms to identify the composition meter, including the beat, bar and line information, and special notes.

The melody data that had been restructured by adding space and comma symbols and special characters were then processed using the bag of words technique, a technique commonly used in Natural Language Processing (NLP) for text generation. Thus, the melody data were arranged into a character-based representation, and the output was a character sequence which represents note sequence, the composition meter and the special notes. The character sequence was then represented using the one-hot encoding technique, and used as input to train the LSTM network.

The melody generation stage consists of two parts, namely note sequence generation and composition meter measurement. The note sequence generated using the LSTM method by implementing note-by-note generation technique was measured using composition meter rules. The resulting note sequence would be declared as a legit melody if it complied with the composition meter rules. Otherwise, the resulting note sequence would be deleted, and the LSTM network will repeat the note-by-note generation.

The note sequence generated by the LSTM network was reversed to the GSPN format in order to measure its composition meter. Based on the experiments that have been carried out, the LSTM network was not always able to produce a note sequence that complied with the composition meter rules in one generation process. However, it took no more than 10 attempts at generation for the LSTM network to successfully produce a note sequence that complied with the defined characteristics. Therefore, the process of generating a single melody was limited to 10 attempts in producing the note sequence. The melody generation process would be declared as a failure if, up to 10 attempts in producing a note sequence, did not meet the composition meter rules. Fig. 2 shows the workflow diagram of the Gamelan melody generation using the LSTM method and the composition meter rules with n representing the numbers of the note sequence generation attempt.

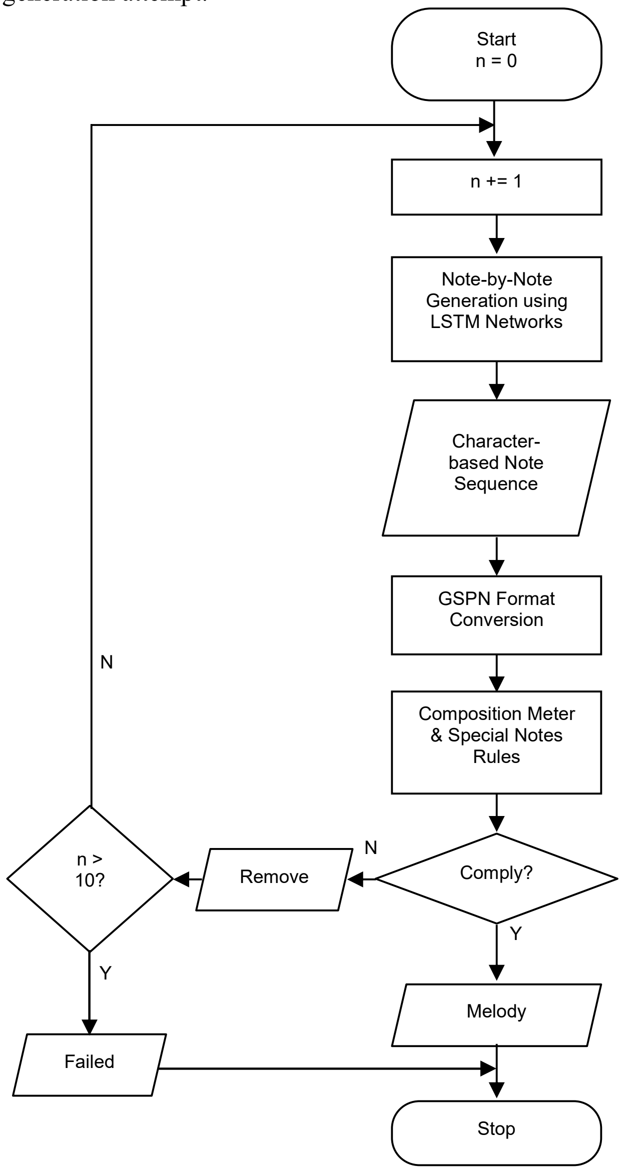


Figure 2. Workflow diagram of the proposed Gamelan melody generation.

The proposed method consists of five stages, which are: 1) GSPN conversion, 2) composition meter measurement and data restructuring, 3) character-based representation, 4) LSTM training, 5) melody generation, including GSPN reversion for the generated note sequence.

A. The GSPN Conversion

Composition data in Gamelan sheet music format were collected from www.gamelanbvg.com. The data consist of 59 *ladrang*-type compositions of the *slendro* musical system and the *manyura* musical mode played in the time signature of 1/2. All sheet music consists of the melody and melodic skeleton, and lyrics. However, the lyrics and the melodic skeleton part were omitted because they were not used for melody generation. Fig. 3 shows an example of the note sequence of melody of the composition entitled *Eling-eling Suralaya*.

Next, the sheet music data were converted to a text-based format using GSPN model. The GSPN model represents a single note unit with four musical elements: note, note register, note duration and legato sign. The *slendro* musical scale system consists of notes 1, 2, 3, 5 and 6. The dotted notes, which represent moments of silence, were converted to the number 0. Thus, the note element consists of 0, 1, 2, 3, 5 and 6. The note register element consists of low notes symbolized by a dot below the note, high notes symbolized by a dot above the note, and middle notes that do not use any note register symbol.

The GSPN model used the code a to indicate the low notes, the code b to indicate the high notes, and no code to indicate the middle notes. The note duration element consists of value of 1, 0.5 and 0.25. The note duration value of 0.5 marked with a single horizontal line above the note is represented using the code A, and the note duration value of 0.25 marked with a double horizontal line above the note is represented using the code B, while the note duration value of 1 that did not use any note duration symbol is represented using no code. The legato sign element used the code x to indicate notes where a legato sign begins, and the code y to indicate notes where a legato sign ends, while no code was used for notes placed between a legato sign and notes that were not part of any legato sign. The melody data were written in the order of note - note register - note duration - legato sign, and in conjunction without spaces. Fig. 4 shows an illustration of melody data conversion into the GSPN format.

In this stage, all sheet music were manually converted to GSPN format using text editor program. The following is an example of sheet music conversion into the GSPN format for a composition shown in Fig. 2.

Ladrang ‘Eling-Eling Suralaya’, laras slendro pathet manyura:
00003b3b3bAx2bAy1bx2b3by2bAx1bAy6Ax1bA2
bB1bB6Ay3Ax5A3y20056x1bA2bAy61bB6B5Ax3
x05A6Ax2Ax5Ay301Ax2A1y6ax0A1A2y00220A2
Bx3By1x02y3303Ax5A3y20056x1bA2bAy6x1bB6
B5Ay3x05A6Ay1b2bx0A3bAy1bAx2bA1by600003
b3b3bAx2bAy1bx02by1bBx2bB6Ay303Ax5A3y20
056x1bA2bAy6x1bB6B5Ay3x05A6Ay2Ax5Ay301
Ax2A1y6a

Manual technique used to convert sheet music to GSPN format is prone to human error in typing the codes. However, the composition meter measurement used to mark the composition meter and special notes, including to verify the melody generated by the LSTM network, can also be used to verify the data in the GSPN format. Therefore, typing errors when converting sheet music into a GSPN format can be identified and corrected. The conversion results were saved into a single TXT document, where each melody data is separated by Enter on a new line.

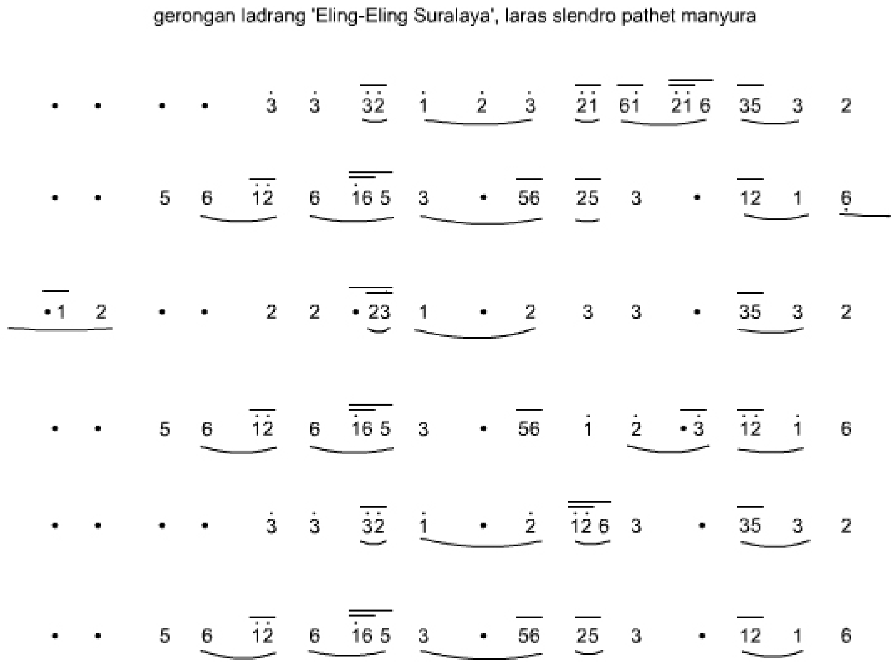


Figure 3. An example of the melody part of a *ladrang*-type composition of the *slendro* musical scale system and the *manyura* musical mode system which is played in the time signature of 1/2.

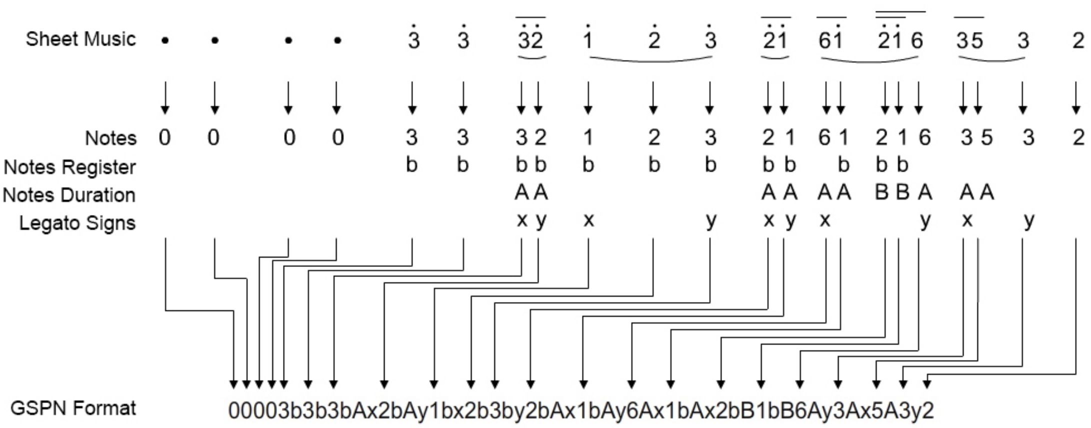


Figure 4. Illustration of text-based conversion in GSPN format.

B. The Composition Meter Measurement and Data Restructuring

Six rules to control the composition meter, including the legato sign, were formalized from the Gamelan sheet music analysis. To be declared as a melody, the note sequence generated by the LSTM network must comply with these six rules. The rules denoted as R1 to R6 are:

R1: The beat duration value is the denominator value of the time signature.

R2: The bar duration value is the beat duration value multiplied by four, where the constant value of four is based on a bar consisting of four beats.

R3: The line duration value is the bar duration value multiplied by two, where the constant value of two is based on a line consisting of two bars.

R4: The minimum number of bars is eight, where the constant value of eight is based on the facts of minimum number of bars found in the dataset.

R5: The total duration of note, bar and line has the same value.

R6: The legato sign has a note as the beginning of the legato sign, and another note as the end of the legato sign.

The dataset used data in the GSPN format which only contains codes of notes, note durations, note registers and legato signs. Meanwhile, using the Eling-Eling Suralaya note sequence data exemplified above, the composition meter data acquisition was performed by first separating the data using commas. The results were then used as input data. The next step is extracting the notes data based on input data. The notes data were used as a reference for the note register, note duration and legato sign data. The note register code used code a and b for high and low note, respectively, while the note that was not followed by these codes is the middle note. The note duration code used code A and B for note duration values of 0.5 and 0.25, respectively, while the note that was not followed by these codes is the note with duration value of 1. The legato sign code used code x and y for the beginning and end of the legato, respectively.

After doing the above steps, the data in GSPN format were extracted into note, note register, note duration and legato sign data, including the note unit (the note and its attribute) as illustrated in Table I, which uses a tabular format to display the data with code - (dash) representing null data.

TABLE I. ILLUSTRATION OF THE GSPN FORMAT DATA EXTRACTION

Note Elements	Note Sequence Index													
Note Sequence	0	0	0	0	3	3	3	2	1	2	3	…	1	6
Note Register	-	-	-	-	B	b	b	b	b	b	B	…	-	a
Note Duration	-	-	-	-	-	-	A	A	-	-	-	…	-	-
Note Legato	-	-	-	-	-	-	x	y	x	-	Y	…	y	-
Note Unit	0	0	0	0	3b	3b	3bAx	2bAy	1bx	2b	3by	…	1y	6a

Furthermore, the note unit duration data were calculated to set the beat, bar and line using a procedure proposed by [23], where the value of a note unit is 0.5 and 0.25 if it contains code A and B, respectively, while the value of a note unit is 1 if it contains no code A or B. The value of each note unit was then calculated based on R1, R2 and R3, respectively. The following is an example of the results of determining beats, bars and lines:

Beat duration : (0, 0), (0, 0), (3b, 3b), (3bAx, 2bAy, 1bx), (2b, 3by), …, (1y, 6a)
Bar duration : (0, 0, 0, 0, 3b, 3b, 3bAx, 2bAy, 1bx), (2b, 3by, …), …, (…, 1y, 6a)
Line duration : (0, 0, 0, 0, 3b, 3b, 3bAx, 2bAy, 1bx, 2b, 3by, …), …, (…, 1y, 6a)

The LSTM network was expected to be able to recognize the special notes that control the Gamelan musical mode system based on the last note in bars and lines. Information of the special notes was added into the data melody using unique letters as well. The codes used any character that differed from each other and from the GSPN codes. The first step was restructuring the note unit data by separating them with a space mark. Next was restructuring the beat data by adding commas to separate all the beats except for the last beat. This is to distinguish between the last beat and previous beats. The last note unit in every bar data, except the last bar, was restructured by adding the code v, and the code was placed before the note unit data. Meanwhile, the last note unit in every line data, except the last line, was restructured by adding the code L, and the code was placed before the note unit data. Melody has a repetition pattern, which repeats from the first note after reaching the last note. Thus, the first and last notes were marked to be recognized by the LSTM network as a repeating pattern. The code S was added to the first note unit of the melody and the code E was added to the last note unit of the melody.

The following is an example of data restructuring by adding codes in special notes, where the codes for special notes are displayed in bold for easy reading, in which the first bar is added with code **v** in its last note, the second bar that defined the first line is added with code Lv in its last note:

(0**S** 0, 0 0, 3b 3b, 3bAx 2bAy **v**1bx), (2b 3by, 2bAx 1bAy 6Ax 1bA, 2bB 1bB 6Ay 3Ax 5A, 3y **Lv**2)

The following is an example of the results of data restructuring for a composition shown in Fig. 3.

0S 0, 0 0, 3b 3b, 3bAx 2bAy v1bx, 2b 3by, 2bAx 1bAy 6Ax 1bA, 2bB 1bB 6Ay 3Ax 5A, 3y Lv2, 0 0, 5 6x, 1bA 2bAy 6, 1bB 6B 5Ax v3x, 0 5A 6Ax, 2Ax 5Ay 3, 0 1Ax 2A, 1y Lv6ax, 0A 1A 2y, 0 0, 2 2, 0A 2Bx 3By v1x, 0 2y, 3 3, 0 3Ax 5A, 3y Lv2, 0 0, 5 6x, 1bA 2bAy 6x, 1bB 6B 5Ay v3x, 0 5A 6Ay, 1b 2bx, 0A 3bAy 1bAx 2bA, 1by Lv6, 0 0, 0 0, 3b 3b, 3bAx 2bAy v1bx, 0 2by, 1bBx 2bB 6Ay 3, 0 3Ax 5A, 3y Lv2, 0 0, 5 6x, 1bA 2bAy 6x, 1bB 6B 5Ay v3x, 0 5A 6Ay, 2Ax 5Ay 3, 0 1Ax 2A, 1y 6aE

The whole process of the data restructuring for the composition meter and special notes was carried out using the MATLAB R2019b program and the results were exported into CSV format, where each line in the CSV document contains data for one melody. The dataset consists of 59 compositions distributed into 59 rows in a single CSV document. Instead of sending the data to be represented in a character-based format, the use of CSV document was chosen to store the data.

C. Character-Based Representation

The CSV document that contains composition data was used as input for the LSTM network. The MATLAB R2019b program was also used to develop the LSTM network. The design is to train the LSTM network using composition data set into a character-based representation format, and Unicode which is a standard for writing text and symbols to be processed using a computer was used to represent the dataset in text format. The code \x0002 (hash mark) was used in the beginning of each line of data to mark the beginning of the text (melody), the code \00B7 (middle dot mark) was used to mark each space in each melody, and the code \x2403 (end of text mark) was used to mark the end of each melody.

The following is an example of the result of representing the data, previously used as an example, into a character-based format.

#0S•0,•0•0,•3b•3b,•3bAx•2bAy•v1bx,•2b•3by,•2bA
x•1bAy•6Ax•1bA,•2bB•1bB•6Ay•3Ax•5A,•3y•Lv2,
•0•0,•5•6x,•1bA•2bAy•6,•1bB•6B•5Ax•v3x,•0•5A•6
Ax,•2Ax•5Ay•3,•0•1Ax•2A,•1y•Lv6ax,•0A•1A•2y,•
0•0,•2•2,•0A•2Bx•3By•v1x,•0•2y,•3•3,•0•3Ax•5A,•3
y•Lv2,•0•0,•5•6x,•1bA•2bAy•6x,•1bB•6B•5Ay•v3x,
•0•5A•6Ay,•1b•2bx,•0A•3bAy•1bAx•2bA,•1by•Lv6
,•0•0,•0•0,•3b•3b,•3bAx•2bAy•v1bx,•0•2by,•1bBx•2
bB•6Ay•3,•0•3Ax•5A,•3y•Lv2,•0•0,•5•6x,•1bA•2bA
y•6x,•1bB•6B•5Ay•v3x,•0•5A•6Ay,•2Ax•5Ay•3,•0•
1Ax•2A,•1y•6aE•

TABLE II. Illustration of the One-Hot Encoding Technique Implementation

F	Melody Data														
	#	0	S	•	0	,	•	0	•	0	,	•	3	b	•	…
#	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
•	0	0	0	1	0	0	1	0	1	0	0	1	0	0	1	…
,	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	…
0	0	1	0	0	1	0	0	1	0	1	0	0	0	0	0	…
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
3	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	…
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
E	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
L	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
S	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	…
a	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
b	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	…
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
x	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…
y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	…

The data of 59 melodies produced a total of 27.048 characters consisting of 19 unique characters used as features, which are: 0 1 2 3 5 6 A B a b x y # •, v L S E. Furthermore, the One-Hot Encoding technique was applied to the input data by assigning a value of 1 to activate a feature and 0 to deactivate the other features. Table II, with column F stands for features, shows an illustration of the implementation of the One-Hot Encoding technique.

D. The LSTM Network Architecture and Training

The LSTM network learnt the melodic pattern, composition meter and special notes using the sequence prediction technique, where the target of the next character is based on the previous character. The exclamation mark was used as the target for the last character in a melody. In other words, the target is an exclamation mark when the sequence (data in a row) reaches the last character, and then ended with the end of text mark. The network architecture uses five layers, which are input, LSTM, fully connected, soft max and output. The input size was 19, the number of hidden units is 200. Loss function was measured using cross entropy. Adam optimization algorithm which is the development of the stochastic gradient descent algorithm [27] was selected to train the LSTM network. The learning rate was 0.01, the size of mini batch was 77 and the number of maximum epoch was 500.

The best prediction was determined based on the highest mini batch accuracy and the lowest error loss. In the LSTM network, input X contains (X_1, X_2, …, X_{end}) where X_n is the element of (0, 1, 2, 3, 5, 6, A, B, a, b, x, y, #, •, ,, v, L, S, E) and X_{end} is total character in dataset numbering 27,048. Meanwhile, output Y contains (Y_1, Y_2, …, Y_{end}), where $Y_n = X_{n+1}$. If input X_n is the character E, output Y_n, is the middle dot mark, and the following input X and output Y are automatically filled with the end of character mark and the exclamation mark, respectively. This indicates that one melody data has been learned, and then input X is the hash mark which is a character that represents the beginning of one melody data. This continued until it reached the last character in the dataset, which is the 27,048th character.

Fig. 5 shows the LSTM network architecture where X represents the input, H represents the hidden layer and Y represents the target which is the next character from the previous character input, as well as an example of the input data for one melody and its target in the LSTM network training, with a hash mark indicating the first input and an exclamation mark indicating the target for the last input. Melody 1 which is represented in 472 characters becomes the first melody input, followed by Melody 2 and so on. All characters had been converted into one-hot encoding format first before they were sent to the networks. The best result of the training accuracy was obtained with a mini batch accuracy of 86.9% with an error loss of 0.0143 at epoch 500. Fig. 6 shows the training progress data in numbers and the curve of training progress.

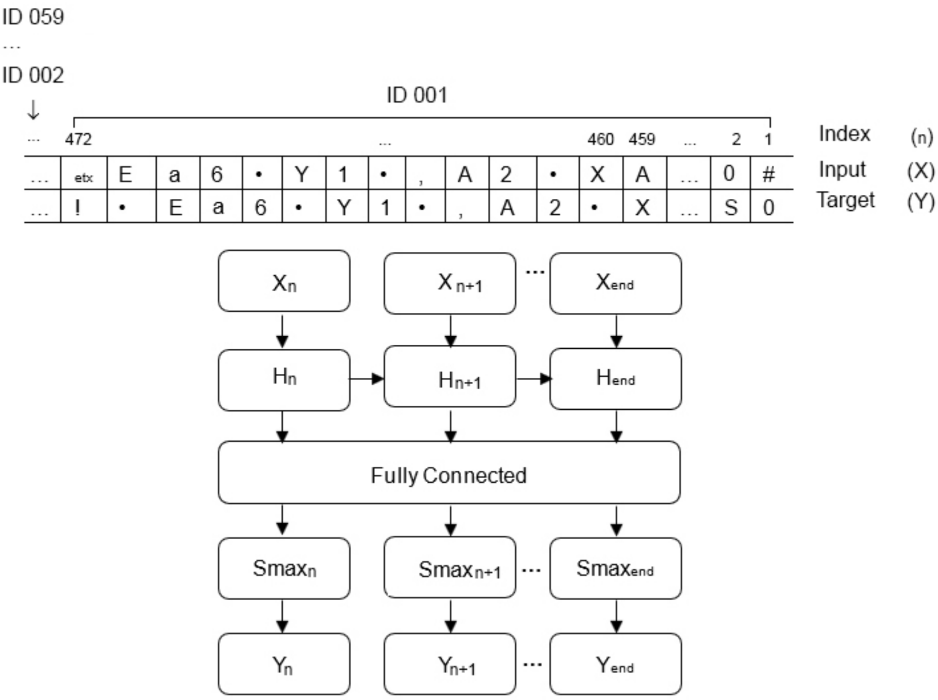


Figure 5. The LSTM network architecture.

Epoch	Iteration	Time Elapsed (hh:mm:ss)	Mini-batch Accuracy	Mini-batch Loss	Base Learning Rate
1	1	00:00:05	35.49%	1.6725	0.0100
100	100	00:06:56	89.39%	0.3051	0.0100
200	200	00:13:57	86.85%	0.0992	0.0100
300	300	00:21:01	86.98%	0.0374	0.0100
400	400	00:28:07	88.26%	0.0198	0.0100
500	500	00:35:17	86.91%	0.0143	0.0100

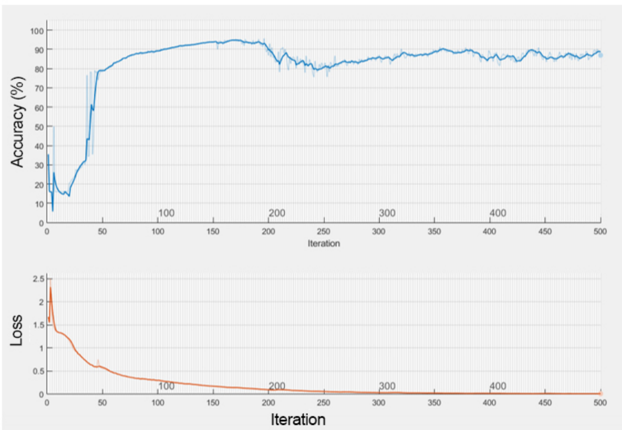


Figure 6. The training progress in numbers (up) and the curve of training progress (buttom).

E. Melody Generation

The melody generation was controlled by two stopping criteria, which are the output prediction result or predicted number of characters. The LSTM network stops predicting when the output is an exclamation mark, or stops predicting when the number of characters of the output results are equal to the maximum number of predicted characters. Determining the maximum number of predicted characters was based on the biggest number of characters of a melody in the dataset. A melody containing 28 bars represented by 808 characters had the biggest number of characters, and the number was then rounded up to 900. Thus, the LSTM network continued to predict a sequence of characters as long as predicted number of characters was less than or equal to 900, and the prediction stopped when the output was an exclamation mark character even though the number of predicted characters had not reached the maximum number of characters to be predicted.

The note-by-note generation technique was implemented in the melody generation phase by generating the sequence of characters as the note sequence in the form of a character-based representation. Once the system was run, it generated character-by-character based on the highest weighted value. Although the mini batch accuracy reached 86,9%, errors in recognizing the melodic pattern, including in complying with the composition meter or the placement of special notes still occurred in the remaining 13.1%. On the other hand, in this proposed method, recognizing the melodic patterns was left to the ability of the LSTM network, while the composition meter and the special notes can be measured by identifying the placement of the v, L, Lv and E codes in the sequence of generated characters. Thus, the sequence of generated characters was converted back into the GSPN format note sequence for measurement of the composition meter. If the result complied with the composition meter rules and the placement of the special notes, the note sequence would be declared as a legit melody. Otherwise, the note sequence would be deleted, and the LSTM network would repeat the generation process. This was performed repeatedly with a limit of 10 generations of experiments. Three melodies were generated in row, and each generated note sequence has complied with the composition meter rules (R1-R6) to be declared as melody in less than 10 generation attempts. The three generated melodies were distributed into two melodies which are eight bars long and one melody is 16 bars long as shown below (the example of the generated melody #1 in the sheet music format can be seen in Fig. 7):

Melody #1 generated in 6 trials:

0S•0,•0•0,•6•6,•0A•6A•v1bx,•0•2by,•3b•3b,•0•1bAx•3bA,•2by•Lv2bx,•1b•0,•6A•1bAy•2bx,•0A•3bAy•1bAx•2bA,•6y•v3,•0•0,•1bAx•2bAy•6x,•0A•5Ay•3Ax•5A,•3y•Lv2,•0•0,•3•2x,•0•6ay,•6aAx•2Ay•v1,•0•0,•3•5x,•0A•6Ay•3Ax•5A,•3y•Lv2,•0•0,•5•6x,•1bA•2bAy•6x,•1bB•6B•5Ay•v3x,•0•5A•6Ay,•2Ax•5Ay•3,•0•1Ax•2A,•1y•6aE•!

Melody #2 generated in 5 trials:

0S•0,•0•0,•0•0,•0•v0,•0•0,•5•6x,•1bA•2bAy•6x,•1bB•6B•5Ay•Lv3,•0•0,•0•0,•0•3,•0•v6x,•0•1by,•0•1bx,•0•2b,•0A•3bAy•Lv1bx,•2b•6,•1b•2b,•0A•3bAy•1bA•2bA,•6•v3x,•0•0A•2A,•3A•5A•6y,•0•3Ax•5A,•3y•Lv2,•0•0,•3•2x,•0•1y,•1Ax•2Ay•v1,•0•0,•3•5x,•0A•6Ay•3Ax•5A,•3y•Lv2,•0•0,•3•2x,•0•1y,•1Ax•2Ay•v1x,•0•2y,•6a•1x,•0A•2Ay•2x,•1A•3Ay•Lv3,•0•0,•0•0,•0•6,•0A•1bA•v2b,•0•0,•3b•5x,•0A•3bAy•1bAx•2bA,•6y•Lv5x,•0•3y,•0•0,•6•6,•0A•6A•v1bx,•0•2by,•2bAx•3bAy•1bx,•0A•2bAy•6Ax•1bA,•6y•Lv5,•0•0,•3Ax•5Ay•6,•0•2x,•0A•3Ay•v1x,•0•2A•3Ay,•3Ax•5Ay•2x,•0A•3Ay•1Ax•2A,•1y•6aE•!

Melody #3 generated in 2 trials:

0S•0,•0•0,•3•3,•0A•3A•v6x,•1b•2bA•3bAy,•3bAx•5bAy•2bx,•0A•3bAy•1bAx•2bA,•1by•Lv6,•0•0,•0•0,•3b•3b,•3bAx•2bAy•v1bx,•0•2by,•6•5x,•0A•3Ay•6x,•1bB•6B•5Ay•Lv3x,•2A•3A•2y,•0•0,•2•2,•0A•2A•v3x,•0•5y,•6•6x,•1bA•2bAy•6x,•1bB•6B•5Ay•Lv3,•0•0,•1b•2bx,•1bA•6Ay•3Ax•5A,•3y•v2,•0•0,•3Ax•5Ay•3,•0•1Ax•2A,•1y•6aE!

V. Results and Discussion

An LSTM network was trained using the sequence prediction technique with the input being in the form of data representing melodic patterns, composition meter, and special notes that determine the musical mode system of compositions. The character-based representation technique commonly used in NLP was implemented as a solution for the system in learning Gamelan melodies. The generated melodies were expected to comply with the characteristics of the melodic pattern, composition meter, and characteristics of the musical scale and musical mode systems. During its training, the LSTM Network was able to achieve the training accuracy of 86.9%. This fact was used to draw the conclusion that the LSTM network can produce legit melodies but inconsistent results in the composition meter and special notes may occur. In other words, based on the proposed melody generation model, the LSTM network cannot always produce a single legit melody in one process, so the LSTM network needs to be given the chance to repeat the melody generation until reaching the legit melody.

The following is an example of a generated melody that does not comply with the composition meter rules. The generated melody complies with almost all of the composition meter rules, i.e., the number of beats is 16 and each beat has the duration value of 2, each bar has the duration value of 8; the number of lines is 2 and each line has the duration value of 16, the first note contains the code S, and the last note contains the code E, including the code v and L, but the number of bars is 4 which is less than 8 as determined by the rules.

(0S 0, 0 0, 6 6, 0A 6Bx 1bBy v5x), (0 6y, 1b 2bx, 0A 3bAy 1bAx 2bA, 1by Lv6), (0 0, 1b 2bx, 1bA 6Ay 3Ax 5A, 3y v2), (0 0, 3Ax 5Ay 3, 0 1Ax 2A, 1y 6aE)

Another example of an error in the melody generation is that the generated melody did not comply with the composition meter rules for the number of beats, and errors in the number of beats created errors in all other elements. Moreover, the generated melody has two beats with the duration value that is not 2. The two beats are written in bold as follows, where notes followed by code A and B have duration 0.5 and 0.25 respectively, and notes not followed by code A and B have a duration of 1:

(0S 0), (0 0), (3b 3b), (3bAx 2bAy v1bx), (2b 3by), (2bAx 1bAy 6Ax 1bA), (2bB 1bB 6Ay 3Ax 5A), (3y Lv2), (0 0), (5 6x), (1bA 2bAy 6), (0 3b), **(Lv2x)**, (0A 3Ay) (1x, 2A) (3Ay Lv6a), (0 0), (0 0), (0 0), (6 v6), (0 0), (3 6x), (0A 1bAy 1bx), (0A 2bAy Lv1b), (0 0), (1b 2b), (0 3bx), (0A 5bAy v2), (0 0), (6 5x), (0A 3Ay 6x), (1bB 6B 5Ay Lv3), (0 0), (2 2), (0 0), (2Ax 3Ay v2x), (0 3y), (6a 1x), (0A 2Ay 2x), (1A 3Ay Lv3), (0 0), (2 2), (0 0), (2Ax 3Ay v2x), **(0)** Instead of increasing the performance of the LSTM network until it can achieve a training accuracy at 100% which, one might say, is almost impossible, the training accuracy that had been achieved by the LSTM network was stated to be sufficient. Therefore, the melody generation system added the composition meter and special notes measurement feature. These features were used to measure the accuracy of the sequence of notes generated by the LSTM network against the characteristics of notes, bars, and lines in the Gamelan melodies. Based on the experiments, the LSTM network was able to generate melodies that complied with composition meter rules in less than 10 attempts.

The dataset which used 59 melodies produced a total of 27,048 characters. Compared with [4] which used the character-based representation technique with 87 unique characters and [19] which used the word-based representation technique with 39 unique words, the approach used in this study had fewer number of features, which is 19 unique characters. On the other hand, the authors of [4, 19] used a larger number of inputs which had 155,222 character and 539,609 word counts, respectively. In general, the greater the number of inputs, the more variable the output. The number of datasets containing 59 compositions can be categorized as minimal or even less, so the similarity among the generated melodies was a consequence. However, character-based representation and one-hot encoding in sequence prediction techniques that predict one single character in each subsequent step seemed to increase variation in melody, although not very significantly. Moreover, the characteristics of Gamelan music allows the similarity among the generated melodies. In fact, according to [28], a legendary Gamelan music figure, it is advisable to create Gamelan music compositions by modifying existing melodies as needed. Despite the problem of similarity in generated melodies, the method proposed in this study can be said to be suitable to overcome the problem of generating Gamelan melodies. The proposed LSTM network was designed to predict each character sequence based on the highest weighted value. It is still up for investigation whether the use of beam search, top-k sampling, and/or nucleus sampling can significantly increase the difference in the generated melodies. In addition, the consequences of the similarity of the resulting melodies could be overcome by increasing the number of datasets to reach hundreds of thousands of characters.

Furthermore, the three generated melodies were sent to three experts who possess more than 15 years of experience in Gamelan music to be assessed for their suitability to the characteristics of Gamelan music. One of the experts has an academic background in Gamelan music from Universitas Negeri Semarang, Central Java, Indonesia. The other two experts are practitioners from *Pangreksa Budaya* Gamelan studio, Semarang, Indonesia, and *Mardayu* Gamelan studio, Surakarta, Indonesia. Three generated melodies in the form of GSPN format were manually converted into sheet music format for aforementioned experts to read, hum and evaluate. A conversion results example is shown in Fig. 7. Experts used the solfeège technique in evaluating the melody performed by humming the melody based on the sheet music. This technique is commonly used by Gamelan experts to recognize patterns in music and gain an understanding of the melody and the composition structure in depth.

0 0, 0 0, 6 6, 0A 6A 1bx, 0 2by, 3b 3b, 0 1bAx
3bA, 2by 2bx, 1b 0, 6A 1bAy 2bx, 0A 3bAy 1bAx
2bA, 6y 3, 0 0, 1bAx 2bAy 6x, 0A 5Ay 3Ax 5A, 3y
2, 0 0, 3 2x, 0 6ay, 6aAx 2Ay 1, 0 0, 3 5x, 0A 6Ay
3Ax 5A, 3y 2, 0 0, 5 6x, 1bA 2bAy 6x, 1bB 6B
5Ay 3x, 0 5A 6Ay, 2Ax 5Ay 3, 0 1Ax 2A, 1y 6a

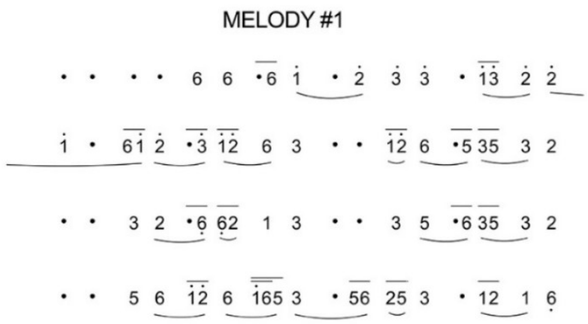


Figure 7. Example of sheet music conversion results.

The experts judged the quality of the generated melodies using a range of values from 1 to 5, corresponding to very unsuitable, unsuitable, quite suitable, suitable, and very suitable. Table III shows the results of the experts' evaluation based on the sound characteristics of Gamelan music with the questions Q1: 'How suitable are the generated melodies in matching the sound characteristics of actual Gamelan melodies?', and Q2: 'How suitable are the generated melodies in matching the sound characteristics of the *slendro* musical scale and the *manyura* musical mode?'

TABLE III. RESULTS OF EXPERT TEST

Experts	Melody #1		Melody #2		Melody #3	
	Q1	Q2	Q1	Q2	Q1	Q2
Expert 1	4	4	4	3	4	4
Expert 2	4	4	4	4	4	4
Expert 3	4	4	4	4	4	4

The results of the experts' evaluation based on the sound characteristics of Gamelan music compositions (Q1) showed that all of the generated melodies could be accepted by all experts with a value of 4 (suitable).

Similar results also applied to the evaluation based on the sound characteristics of the *slendro manyura* melodies (Q2). All melodies except melody #2 were rated 4 (suitable) by all the experts. The melody #2 was rated 3 (quite suitable) by expert 1. Melody #2 contains 16 bars or twice as long as Melody #1 and Melody #3 which are eight bars long. Based on the judgment of expert 1, the 12th bar in the 6th bar-line has a note sequence reducing the sense of the *manyura* mode system. Furthermore, it was suggested to change the last note in the 12th bar, from 5 to 3, so that the sense of the *manyura* musical mode can still be achieved on the line. However, these suggestions affected pitch shifts and changed pitch elements as illustrated in Fig. 8.

It is interesting that two other experts can accept Melody #2 with similar suggestion, but gave higher rate for Q2. It can be concluded, then, that Melody #2 has complied with the sound characteristics of Gamelan melodies including its musical mode system with minor corrections. Considering that Melody #1 and Melody #3 have a length of 8 bars which is half of the length of the bars in Melody #2, the proposed method still needs to be improved so that it will be able to produce melodies with a length of more than 8 bars. Melody fragmentation and links between fragments can be implemented to generate long melodies.

Original generated line

0 0	3b 5x	0A 3bAy 1bAx 2bA	6y 5x

Expert 2' suggestions:

0 0	3Ab 5Abx 3b	1bAx 2bAy 1bA 6Ax	0A 5Ay 3

Figure 8. Expert suggestions.

VI. Conclusion and Future Work

This study proposes a Gamelan melody generation system that involves the recognition of melodic patterns, composition meter rules, and special notes which represent ambiguous rules in determining the musical mode system. The LSTM method was chosen to find a solution for these three characteristics. An LSTM network was built and trained using the sequence prediction technique. The inputted data are composition data in sheet music format which are in turn converted into GSPN format, a Gamelan music notation writing system based on ABC notation. Data in the GSPN format are added with codes representing the composition meter and special notes. Next, the data are restructured into a character-based representation format to be used as input for the LSTM network. The LSTM network training showed decent results, but the LSTM network was not always able to generate melodies that are on target in one generation process. However, it took no more than 10 attempts for the LSTM network to generate legit melodies. Overall, the expert test results show that the generated melodies can comply with the characteristics of Gamelan music and the targeted musical scale and musical mode systems.

After the experts' evaluation, the generated melodies were played by three musicians who play three *Gamelan* instruments and one singer. The melody was sung based on the pronunciation of the notation in the Javanese, where the notes 1, 2, 3, 5, and 6 are read in Javanese language with *ji, ro, lu, mo* and *nem*. The performance can be seen in the YouTube channel with the link can be found in the end of this section.

There is still more work to be done in order to perfect the Gamelan melody generation system. The conversion of the generated melody to sheet music with a manual technique needs to be replaced with an automatic conversion technique before the system is implemented to the user, such as in a work of The Bach Doodle with its sheet music-based user interface by [26]. Moreover, the LSTM network performance in generating melodies with controllable composition meter and user preferences features still needs to be improved. The work that needs to be done in the near future is to publish the Gamelan melody generation system on the Internet so that it can be accessed widely and publicly.

Data Availability

The dataset used in the experiment can be accessed through the following link:

https://drive.google.com/drive/folders/1E1ZZ_pWjBW1XRNYzVIBaaI2cgaOxFmxv?usp=sharing

An example of generated melody performed by Gamelan singer and musicians can be seen in the YouTube channel through the following link:

https://www.youtube.com/watch?v=kK6F9a3q--Q

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

A. M. Syarif conducted the research; S. Suprapto analyzed the data; A. Azhari supervised the LSTM method implementation, K. Hastuti as the knowledge engineer, A. M. Syarif and K. Hastuti wrote the paper; all authors had approved the final version.

References

[1] Y. Huang, X. Huang, and Q. Cai, "Music generation based on convolution-LSTM," *Computer and Information Science*, vol. 11, no. 3, pp. 50-56, 2018.

[2] H. Mukherjee, A. Dhar, M. Ghosh, *et al.*, "Music chord inversion shape identification with LSTM-RNN," *Procedia Computer Science*, vol. 167, pp. 607-615, 2020.

[3] Y. Yan, E. Lustig, J. VanderStel, *et al.*, "Part-invariant model for music generation and harmonization," in *Proc. 18th International Society for Music Information Retrieval Conference*, Suzhou, China, 2018, pp. 204-210.

[4] P. S. Rani, S. V. Praneeth, V. R. K. Reddy, *et al.*, "Music generation using deep learning," *International Research Journal of Engineering and Technology*, vol. 7, no. 3, pp. 4937-4939, 2020.

[5] B. Royal, K. Hua, and B. Zhang, "Deep composer: Deep neural hashing and retrieval approach to automatic music generation," in *Proc. IEEE International Conference on Multimedia and Expo*, London, United Kingdom, 2020, pp. 1-6.

[6] M. Dua, R. Yadav, D. Mamgai, *et al.*, "An improved rnn-lstm based novel approach for sheet music generation," *Procedia Computer Science*, vol. 171, pp. 465-474, 2020.

[7] J. P. Briot, "From artificial neural networks to deep learning for music generation: history, concepts and trends," *Neural Computing and Applications*, no. 1, 2021.

[8] J. D. Fernandez and F. Vico, "AI methods in algorithmic composition: A comprehensive survey," *Journal of Artificial Intelligence Research*, vol. 48, no. 1, pp. 513-582, 2013.

[9] J. Wu, X. Liu, X. Hu, *et al.*, "PopMNet: Generating structured pop music melodies using neural networks," *Artificial Intelligence*, vol. 286, pp. 1-15. 2020.

[10] F. Liang, M. Gotham, M. Johnson, *et al.*, "Automatic stylistic composition of bach chorales with deep LSTM," in *Proc. 18th International Society for Music Information Retrieval Conference*, Suzhou, China, 2017, pp. 449-456.

[11] H. Eedi, "Artificial music generation using LSTM networks," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 2, pp. 4315-4319, 2019.

[12] S. Mangal, R. Modak, and P. Joshi, "LSTM based music generation system," *International Advanced Research Journal in Science, Engineering and Technology*, vol. 6, no. 5, pp. 47-54, 2019.

[13] H. H. Mao, T. Shin, and G. W. Cottrel, "DeepJ: Style-Specific music generation," in *Proc. 12th IEEE International Conference on Semantic Computing*, California, USA, 2018, pp. 377-382.

[14] B. Raja, G. Vignesh, G. Sandhya, *et al.*, "LSTM based approach for generating music from MIDI notes," *International Journal of Advanced Science and Technology*, vol. 29, no. 9s, pp. 2745-2752, 2020.

[15] H. W. Dong, W. Y. Hsiao, L. C. Yang, *et al.*, "MuseGAN: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment," in *Proc. the Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, 2018, pp. 34-41.

[16] K. Hastuti, A. Azhari, A. Musdholifah, *et al.*, "Rule-based and genetic algorithm for automatic gamelan music composition," *International Review on Modelling and Simulations*, vol. 10, no. 3, pp. 202-212, 2017.

[17] M. Bretan, G. Weinberg, and L. Heck, "A unit selection methodology for music generation using deep neural networks," in *Proc. 8th International Conference on Computational Creativity*, Atlanta, 2017, pp. 72-79.

[18] C. Z. A. Huang, A. Vaswani, J. Uszkoreit, *et al.*, "Music transformer: Generating music with long-term structure," in *Proc. 7th International Conference on Learning Representations*, New Orleans, 2019, pp. 1-15.

[19] K. Choi, G. Fazekas, and M. Sandler, "Text-based LSTM networks for automatic music composition," arXiv:1604.05358 [cs.AI], 2016.

[20] J. Wu, C. Hu, Y. Wang, *et al.*, "A hierarchical recurrent neural network for symbolic melody generation," *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2749-2757, 2019.

[21] Y. Yu, A. Srivastava, and S. Canales, "Conditional lstm-gan for melody generation from lyrics," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 17, no. 1, pp. 1-20, 2021.

[22] S. Hastanto. The concept of *pathet* in central Javanese gamelan music. *Durham theses, Durham University*, 1985. [Online]. Available: http://etheses.dur.ac.uk/1214/

[23] N. Ishida, "The textures of central Javanese gamelan music: Pre-notation and its discontents," *Bijdragen Tot De Taal-, Land- En Volkenkunde*, vol. 164, no. 4, pp. 475-499, 2008.

[24] J. Becker, *Traditional Music in Modern Java: Gamelan in a Changing Society*, Honolulu: University of Hawai'i Press, 1980.

[25] R. Supanggah, *Bothekan-garap: The Rich Style of Interpretation in Javanese Gamelan Music*, Surakarta: ISI Press, 2011, ch. VII.

[26] A. M. Syarif, A. Azhari, S. Suprapto, *et al.*, "Human and computation-based musical representation for gamelan music," *Malaysian Journal of Music*, vol. 9, pp. 82-100, 2020.

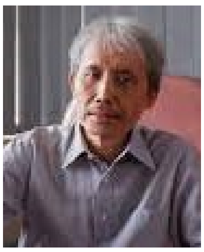
[27] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd International Conference on Learning Representations*, California, 2015, pp. 1-15.

[28] Martopangrawit, *Pengetahuan Karawitan I*, Surakarta: ASKI Press, 1969, pp. 59-60.

[29] C. Z. A. Huang, C. Hawthorne, A. Roberts, *et al.*, "Approachable music composition with machine learning at scale," in *Proc. 20th International Society for Music Information Retrieval Conference*, London, 2019, pp. 793-800.

Arry Maulana Syarif is a lecturer in Faculty of Computer Science, Universitas Dian Nuswantoro, Indonesia. He obtained his Doctor degree in Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Indonesia. His current research interests focus on the Javanese cultural heritage preservation based on Artificial Intelligence and machine learning methods.

Azhari Azhar obtained his Doctor degree in Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Indonesia. He is currently serving as the head of the Computer Science Bachelor Program of Universitas Gadjah Mada. His research interests are Natural Language Programming, Multiagent System, Autonomous System, Digital Game and Development.



Suprapto Suprapto obtained his Doctor degree in Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Indonesia. He is currently serving as the head of the Computer Science Doctoral Program of Universitas Gadjah Mada. His research interests are graph theory, machine learning and data mining.

Khafiizh Hastuti is a lecturer in Faculty of Computer Science, Universitas Dian Nuswantoro, Indonesia. She obtained her Doctor degree in Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Indonesia. She is the founder of Gamelan Research Project, an institution that is active in research of Gamelan and ICT.