Creativity in machines: Music composition using artificial intelligence that passes Lovelace 2.0 test

Shagaf Hasnain[1], Palash Goyal[2],.Rishav Kumar[3]

[1]Gazelle Information Technologies, Delhi, India [2]Mount Carmel School, Delhi, India [3]Mount Carmel School, Delhi, India

[1]shagafhasnain0599@gmail.com [2]palashgoyal1608@mountcarmeldelhi.com

Creativity in machines: Music composition using artificial intelligence that passes Lovelace 2.0 test

Shagaf Hasnain[1], Palash Goyal[2],.Rishav Kumar[3]

[1]Gazelle Information Technologies, Delhi, India [2]Mount Carmel School, Delhi, India [3]Mount Carmel School, Delhi, India

[1]shagafhasnain0599@gmail.com [2]palashgoyal1608@mountcarmeldelhi.com

Abstract--In this paper, the unexplored area of infusing consciousness and creativity are explored. The existing framework developed by the authors is being deployed by using different AI techniques to train the model on musical data sequences with different compositions so the model will be able to generate musical notes of various moods. The proposed system is based on PALASH 1.0 framework[1], which generates different moods of music which is like human-created compositions. These AI-generated music are produced using the creativity of machines in such a way that it passes the Lovelace 2.0 test[2].

Index Terms—AI, RNN, Char-RNN, LSTM, MIDI.

Introduction:

Music generation using AI is being researched for a long time and a lot development has been done in this area. But these developments are limited to the generation of simple musical notes. The music generated lacks the mood detection that is the emotion meaning which can be easily detected in a human-composed music. With this paper we are proposing a system based on the framework PALASH 1.0 which we suggested in our earlier paper[1], it was an approach to enhance the creativity of machines based on the earlier work of Turing's and Riedl[2][3]. This framework uses deep learning techniques to generate music based on different moods/genres.

From the last few centuries, researchers are using mathematical techniques to generate music [4][5]. Music is a sequence of elements (or sound), this was first mentioned by Iannis Xenakis in the early 1950s His music, popularly known as 'Stochastic Music' [6][7] was composed using the concepts of Statistics and Probability. Later the 'IILIAC Computer' which is the best know work of Hiller and Isaacson's [8] that generates music using the 'generate and test' approach. The generated notes were tested first by heuristic compositional rules of classical harmony and only the notes that pass the test are kept.

'CONCERT' is one of the earliest designed generative models which was architectured to compose simple melodies [9]. The model though had some limitations as it was unable to capture the structure of music that was used globally. Later it was observed that sequential modeling techniques like Markov chains or Recurrent neural networks are the most prevailing methods to be used to create models that can learn probable transitions of notes in the given class of music [10][11]. In the last few years, researchers have proposed a lot of new deep neural network models for the generation of music[12][13][14][15][16]. These AI models assign a certain probability to every piece of music and also captures the uniformities in a class of music whether in terms of genres, style, category, etc. Music consists of emotions (or moods) that can be impacted by attributes like tempo, timbre, harmony, loudness, etc.

In our approach, we are taking some existing music data to train our model using these existing data. The model will understand and learn the patterns in music. It will get trained on different moods and compositions of music to be able to understand and differentiate be/tween the different classes of music. Once the model gets trained, it should be able to generate a new sequence of music. It will not just copy-paste the sequence from the training data instead it will understand the patterns and the different genres of music from the training datasets to generate new music.

We will pass our model generated output through Lovelace 2.0 [3] test to see if the model generated quality music that passes the test.

Methodology:

First, we have to represent the music in form of a sequence of events as we are using RNN which takes input in form of sequences. Representation of music can be done in three forms:

- Sheet Music: Pictorial representation of music is known as Sheet music in which a sequence of musical notes is represented.
- ABC Notation: In the ABC-notation there are two parts, the first part represents the metadata and the second part represents the tune which is a sequence of characters representing musical notes
- MIDI: MIDI represents a series of messages such as ‘note on’, ‘note off’, ‘pitch bend’, etc. which is implied by MIDI instruments to generate music

Here, we are representing our music in form of ABC-notation. These musical notes will then be segmented in different classes of music such as happy, sad, soul, etc.

Data Analysis

We have taken different classes of musical data as a source to train our model. We then represent these in form ABC-notation. In fig. 1, some sample musical data represented in form of ABC-notation can be seen. In the first part, it provides metadata to understand the tunes such as (X:), the title (T:), the time signature (M:), the default note length (L:), the type of tune (R:) and the key (K:). In the second part, the tune is given which is a sequence of characters where each character represents some musical note.

```
S:EF
M:4/4
K:A
M:6/8
P:A
f|"A"ecc c2f|"A"ecc c2f|"A"ecc c2f|"Bm"BcB "E7"B2f|
"A"ecc c2f|"A"ecc c2c/2d/2|"D"efe "E7"dcB| [1"A"Ace a2:|
 [2"A"Ace ag=g||\
K:D
P:B
"D"f2f Fdd|"D"AFA f2e/2f/2|"G"g2g ecd|"Em"efd "A7"cBA|
"D"f^ef dcd|"D"AFA f=ef|"G"gfg "A7"ABc |1"D"d3 d2e:|2"D"d3 d2||
```

Fig 1: Sample Training data

To train our model, we will create batches of data and then feed these batch of sequences into our model. Each of these unique characters is assigned some numerical index value. These unique characters will be store based on their emotional moods. A dictionary is created that stores these unique indices and moods as the value and the key is the identified unique characters.

Fig 2: Batches of Data

	Batch-1	Batch-2	...	Batch-150	Batch-151
0	0...63	64...127	...	9536...9599	9600...9663
1	9701...9764	9765...9829	...	19237...19300	19301...19364
.	...	...	...	...	...
.	...	...	...	...	...
.	...	...	...	...	...
14	135814...135877	135878...135941	...	145350...145413	145414...145477
15	145515...145578	145579...145642	...	155051...155114	155115...155178

Then we will feed these batches into our RNN models. Here we are using Many to Many RNN, which gives output equals to the number of inputs. It takes both current and the previous output as input. So, for the first iteration, we will feed zero/dummy input as the previous output along with our input data.

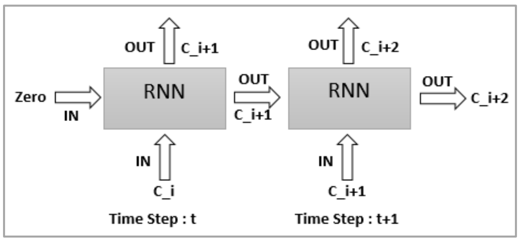

Fig 3: Many to Many RNN

As our model gets trained on these data, we will give some random character to the model from the unique characters that have been identified during training. The model will generate the sequence of characters automatically based on its learning from the training phase.

Techniques Involved:

Char-RNN is a type of RNN with character-based learning that predicts the next character given a sequence of characters [17][18]. Using char-RNN has two benefits, first, the form of the text representation of music has no constraints, and second, it has fewer number of states i.e. a decreased vocabulary which is a drawback of word-based learning methods. Here we have given a sequence of such characters as an input to train our model. Suppose we have a sequence of music as [d, a, e, o, b, a, p, …]. Now we will give ‘d’ as input the model and expect ‘a’ as the output, then we will give ‘a’ and expect ‘e’ as an output, then again we give ‘o’ as input and expect ‘o’ as the output and so on. We train our model in such a way that it will output the next character in the sequence. It will learn these whole sequences and identify the patterns and will be able to generate a new sequence on its own

Despite the widespread use of RNNs, it has certain limitations like vanishing gradients and long-term dependency issues. These issues were resolved by LSTM which uses addition operations and allows the gradient to flow by a separate path [19]. In this approach, we are using three RNN layers each having 256 LSTM units. At each time step, the output generated from all the LSTM units will be given as input to the next layers and the same output will be again given as an input to the same LSTM unit. After these three layers of RNNs, we added a 'Time Distributed' dense layer with Softmax activation in it.

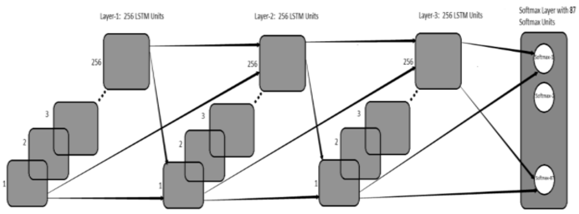

Fig 4. Model Architecture

Softmax is a type of logistic regression that normalizes an input value into a vector of values following a probability distribution whose total sums up to 1. Decimal probabilities are assigned to each class in a multi-class problem. It is implemented using a neural network just before the output layer. The number of nodes must be same in both the Softmax layer and the output layer.

Result:

At first, the model was trained with a learning accuracy of 82 % which is not very good for melodious music generation. To increase the accuracy, we transferred the learning of our previous model and again trained our model again with two-extra layers of LSTM units. Now the learning accuracy of our model has increased to 91%.

After we have trained our model and found the most effective weights, now our model will be predicting and generating music. For prediction, we will provide input from any of the 87 unique characters identified during the training process and the emotional mood (happy, sad, joy, etc.). The model will generate 87 probable values from as output through the Softmax layer. From these returned values, the model will again choose the next character probabilistically and finally, the chosen character will again be fed back to the model and so on. This process continues and keeps on concatenating the output characters to generate music of the given length.

```
MUSIC SEQUENCE GENERATED:

"(37)"E2E D2)|"Am"E2c "G7"B=GB|"Am"D2c cBc|
"D"d2A ABd|"G"g2d e2G|"Am"B2A "D7"A2G|"G"G3 -G2:|
P:B
|:A|"G"BGD "D7"G2A|"G"BGB dBd|"C"efg "B7"b2g|"Em"gfe "Am"dBG|
"D"DEF AGF|"G"GBd g2a|"G"g2d "D7"c2B|"G"G3 -G2:|
P:B
d|"G"dBd gfg|"C"e^de g2e|"F"dBA "D7"ABA|"G"G3 G2:|
```

Fig 5. Generated Musical Sequence represented in ABC-notations

Conclusion:

In this paper, we proposed a system that is able to generate music by giving a set of characters and an emotional mood. The generated output is more creative than the earlier models [11][20][21], in terms of generating mood-based music. We passed the generated output through Lovelace 2.0 test[3] to check the accuracy of our system.

We have four input parameters for our model the epoch weight, the initial character, the length of the music sequence, and the mood based on which the model will generate the musical sequence. Below is the table shows the Lovelace 2.0 test[3] output.

Table 1: Results of Lovelace 2.0 for generated outputs

<table>
<tr><th>Input</th><th>Generated Output</th><th>Passed</th><th>Not Passed</th></tr>
<tr><td>50,45,350</td><td>K:C
|:c2e gee|g2f e2c|"G"d2d d2B|"Am"ABA "D7"B2A|
"G"G3 "G"gfg|"D"f2e "D/f+"def "G"g2f|"Em"e2d
"D"d2c|"Em"B2E "A7"EFE|
"D"D3 "A7"F2E|"G"D2D D2B|"Am"A2G
"D"FED|"G"G3 G3|
"G"ded dcB|"D7"A2D DEF|"G"G3
"D7"FGA|"G"B2G d2B|
"G"ded Bdd|"G"B2B B^AB|"G"d2d
"G7"e2d|"C"c2c "G"B2B|\
"Am"c2B "D7"A2G|
"G"B2G GAB|"C"c2c edc|"G"B2G
"D"A^GE|"G"G3 -A2:|</td><td>✓</td><td>✗</td></tr>
<tr><td>30,78,250</td><td>M::/8
K:D
|:d|"D"d2f "A7"edc|"G"d2d d=cB|"Am"ABA
"D7"ABc|"G"d2G "D7"AGF|"G"G3 G2:|
P:B
g/2g/2|"C"agg g2e|"C"c3 -d2d|(A7"e3 g2e|"D7"d3
d3|"G7"g3 g3|"C"e2d cde|"G"d2B d2^c\
|
"G"D2b "C7"g2e|"G"d3 d2e|"C"f2f
"G"dcB|"D7"c2d e2d|"G"B2G BAG|
"C"c3 ccB</td><td>✓</td><td>✗</td></tr>
<tr><td>80,86,500</td><td>"Em"e3 "Bm"d3|\
"C"cde "G"dcB|"D7"A3 "G"G3||</td><td>✓</td><td>✗</td></tr>
<tr><td>60,59,400</td><td>M:6/8
K:Em
E|"Am"A2B c2d|eef e2d|"G"G2B
"G7"GAB|"C"c6|G2A A3|
"F"c3 AFA|"G7"G2A B2d|"C"e2c GAB|"G"def
g2B|
"Am"efg "D"f2d|"Am"cBA "E7"BAG|"Am"cAA
A2B|"Am"cAA ABc|"E7"e3 -e2d|"G"dBG
GBd|"G"gdB "D7"AGF|"G"GGB d2::
g|"F"a2f "C"g2e|"G"dBG "C"e3|"G"dBG
"D"A2B|"Em"GEE "Em"GEE|
"Bm"DFD FED|"E"B,2B B2A|"Em"G2A BA:|</td><td>✗</td><td>✓</td></tr>
<tr><td>40,56,200</td><td>FC|"Dm"F2F "A7"GFG|"D"A2A
"G"BAG|"C"G2A "D"B2A|
"Am"A2G "D7"DEF|"G"GAB "D"D2A|"G"B2G
BAG|"A7"AGA "D"AGF|"G"G3 -G2:|
P:B
|:b|"G"d^cd def|"G"g2d cB'2f|"C"e2c e2c|"G"B2B
BAG|"G"GAG g2f|
"C"g2G g2G|</td><td>✓</td><td>✗</td></tr>
</table>

Ideas for Future Improvements:

Although we are able to generate music using the creativity of the machine, there is a huge scope of improvement possible in the future.

Further enhancement of creativity and consciousness in machines can be done by training the model on metadata and data which is as per the human behavioral parameters. These systems can be helpful in various areas like education, law and order, healthcare.

Acknowledgment:

We would like to thank Mr. Kirit, the CEO of Gazelle Information Technologies PVT LTD, for his expert advice and a supply of required resources for the implementation of this project.

References:

[1] *Lalit Kumar, Palash Goyal, Rishav Kumar.* Creativity in machines: Music composition using artificial intelligence. *Asian Journal of Convergence in Technology ISSN NO*: 2350-1146 I.F-5.11, (2020)

[2] *A. M. Turing*. Computing machinery and intelligence. *Mind, New Series, Vol. 59, No. 236 (Oct., 1950), pp. 433-460*, (1950)

[3] *Mark O. Riedl*. The Lovelace 2.0 Test of Artificial Creativity and Intelligence. *arXiv:1410.6142v3*, (2014)

[4] *Kirchmeyer, H.* (1968). On the historical constitution of a rationalistic music (Vol. 8, pp. 11{24). Die Reihe

[5] *Lejaren Hiller and Leonard M. Isaacson.* Experimental Music: Composition with an Electronic Computer. New York: *McGraw-Hill*, 1959.

[6] *Gareth E. Roberts* Composing with Numbers: Iannis Xenakis and His Stochastic Music *Math/Music: Aesthetic LinksMontserrat Seminar Spring 2012* March 2, 2012

[7] The Origins of Stochastic Music, Translated by *G. W. Hopkins* from *Xenakis's paper* 'Les Musiques formelles' in *Revue Musicale No.* 253/254 (1966)

[8] *L.Hiller, and L.Isaacson*. Musical composition with a high-speed digital computer (1958). Reprinted in Schwanauer, S.M., and Levitt, D.A., ed. Machine Models of Music.9-21. *Cambridge Mass: The MIT Press*,1993.

[9] *Walter Schulze and Andries Van Der Merwe*. Music generation with Markov models. *IEEE Multimedia*, 18(3):78–85, 2011. ISSN 1070986X. doi: 10.1109/MMUL.2010.44.

[10] *Judy A Franklin*. Recurrent neural networks for music computation. *INFORMS Journal on Computing*, 18(3):321–338, 2006.

[11] *François Pachet and Pierre Roy*. Markov constraints: steerable generation of markov sequences. Constraints, 16(2):148–172, 2011.

[12] *Mason Bretan, Gil Weinberg, and Larry Heck*. A unit selection methodology for music generation using deep neural networks. arXiv preprint arXiv:1612.03789, 2016

[13] *Li-Chia Yang, Szu-Yu Chou, Yi-Hsuan Yang* MIDINET: A Convolutional Generative Adversarial Network for Symbolic-Domain Music Generation arXiv:1703.108472v2, 2017

[14] *Darrell Conklin* Music Generation from Statistical Models. In Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences, Aberystwyth, Wales, 30–35, 2003.

[15] *Keunwoo Choi, George Fazekas, and Mark Sandler* Text-based LSTM networks for Automatic Music Composition arXiv:1604.05358v1, 2018

[16] *Ramón Lopez De,* Mantaras Making Music with AI: Some examples Proceedings of the 2006 conference on Rob Milne: A Tribute to a Pioneering AI Scientist, Entrepreneur and Mountaineer, 2006

[17] *Pratheek I and Joy Paulose* Prediction of Answer Keywords using Char-RNN. *International Journal of Electrical and Computer Engineering (IJECE)* Vol. 9, No. 3, June 2019

[18] *Sutskever, I., Martens, J., Hinton, G.E.:* Generating text with recurrent neural networks. In: *Proceedings of the 28th International Conference on Machine Learning (ICML-11).* pp. 1017–1024 (2011)

[19] *Hochreiter, S., Bengio, Y., Frasconi, P., Schmidhuber, J*.: Gradient flow in recurrent nets: the difficulty of learning long-term dependencies (2001)

[20] *Andrés E Coca, Roseli AF Romero, and Liang Zhao.* Generation of composed musical structures through recurrent neural networks based on chaotic inspiration. *In Neural Networks (IJCNN), The 2011 International Joint Conference on, pages 3220–3226. IEEE*, 2011.

[21] *Eck, D. & Schmidhuber, J. (2002). Arst look at music composition using LSTM recurrent neural networks. Istituto Dalle Molle Di Studi Sull Intelligenza Articiale.*

Biographies:

Shagaf Hasnain is an associate consultant with Gazelle Information Technologies, New Delhi. She holds a bachelor’s in technology with special interests in AI and using AI for practical applications.

Palash Goyal is currently a student of class XII at Mount Carmel School, Sector 23, Dwarka, New Delhi. He started coding at an early age and is currently working on Python, Automation, and Image Processing. It was his personal experience at one of his previous schools, that led to the birth of this idea last year, which he converted into reality with the help of other authors.

Rishav Kumar is currently a student of class XII at Mount Carmel School, Sector 23, Dwarka, New Delhi. His areas of interest include AI and internet of things.