Effectiveness of Markov chain based lyrics in music

1st Prithvi kishan
Department of Electronics and Telecommunication, BMSIT&M
Bangalore, India
prithvi.kishan@yahoo.in

2nd Dhanya Y
Department of Electronics and Telecommunication, BMSIT&M
Bangalore, India
dhanyakarle@gmail.com

3rd S Harshatha
Department of Electronics and Telecommunication, BMSIT&M
Bangalore, India
Harshatha19@gmail.com

4th Saritha I G
Department of Electronics and Telecommunication, BMSIT&M
Bangalore, India
saritha.i.g@bmsit.in

***Abstract*—Several music writers and artists face the challenge of finishing a project they once started. This is mainly attributed to the artists spending large amounts of time finding a suitable flow, melody, or tune for their songs. Our research is aimed at studying the scope and effectiveness of music with varying levels of lyrical meaning in helping artists to come up with ideas and ultimately expedite their music production process. A Random Song Generator (RSG) was developed to generate unique songs composed of vague lyrics mixed with background tracks for holistic integration. This program was distributed to over 300 artists from four genres: rock, pop, rap, and classical music, and an experience of 0 to 5 years in music production. After three months, the artists were surveyed for their RSG usage and the data showed that over 60% of the artists with less than a year of experience used at least one portion of a song generated by the RSG for their composition(sampling). The survey also indicated that all genres had significantly lower music production timelines when sampling from the RSG. The sampling trends among cohorts of different age groups are also discussed. Another objective of this study is to investigate the improvement methods and scalability of the RSG.**

Keywords— Markov chain, Song generator, Lyrics generator, Information theory, Music, Psychology, Artists, Background music, Music genre.

I. INTRODUCTION

Artists and musicians take anywhere between 3 months to a couple of years to produce an entire album. This is because the creative nature of any art, time constraints, and a rushed approach are detrimental to the result of any artist's project [13]. One method to overcome this compromise between creativity and speed is to provide large amounts of creative and unique inputs in a short time. A random song generator produces a near-endless amount of songs with unique lyrics, melody, and flow. The uniqueness of a song is determined by its holistic integration of lyrics and melody, and not just its components [4]. Studies [2,11] found that comprehension and cognitive abilities improved when a non-lyrical music track was played in the background as compared to a lyrical track. [12] demonstrated that song lyrics are becoming simpler over time, hence a song generation model with less priority to lyrical complexity would be beneficial. Hence in our research, we test the effectiveness of a song generator that composed lyrics with varying degrees of meaning and context. Artists are expected to just capture the flow, rhyme scheme, or tune and not the lyrics themselves and hence build on the said components of music along with their lyrics.

A Markov chain is a probabilistic model describing a sequence of possible events in which the probability of a future event is based only on the current state and not the previous states. The Markov Chain model in discrete time follows the given condition(transition probabilities):

$$P(Xn+1 = s|X0 = x0,\ldots,Xn = xn) = P(Xn+1 = s|Xn = xn). \quad (1)$$

The Markov chain is a suitable song generation model because a probabilistic approach can produce vague and even meaningful songs, and it is linguistically restricted. The words from the artist's lyrics are the vertices of our Markov chain, and their probability (weights) is assigned according to their frequency. An excerpt from Pink Floyd's Echoes is represented as a Markov chain as shown in Figure 1.

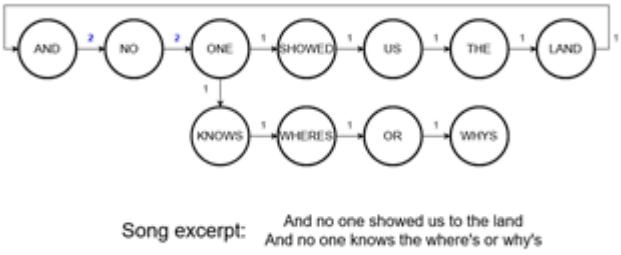

Fig. 1 Markov chain of a song excerpt from Pink Floyd's Echoes.

II. METHODOLOGY

A. Obtaining lyrics and database creation

Pre-existing song lyrics were sourced from the Genius API in python. A minimum of 10 songs were retrieved from each artist which formed the base of their lyrical dictionary. The 15 most popular artists were selected from each of the four genres and a database of their lyrics was created. Similarly, a database of background tracks from various genres was also created.

B. Markov chain based lyrics generation

The lyrics were composed by choosing a random initial word from an artist's lyrics database. Further, we moved across

the Markov chain based on the weights(frequency) of the words which then formed the subsequent words in the lyrics as shown in the Appendix. The maximum number of words parameter is set to 300 words. The total number of lyrics files(NLF) that can be generated is a function of the number of artists(A) and the total number of words in an artists lyrics database and is formulated as:

$$NLF = (A! \times W) / 300 . \quad (2)$$

C. Text-to-speech conversion and audio mixing

A text-to-speech engine is used to vocalize the generated lyrics, and the vocal track is mixed with an instrumental track from its database. The mixing is performed separately in two programming environments, MATLAB [9] and Python [8]. The total number of songs(N) that can be generated from the RSG is:

$$N = NLF \times B \quad (3)$$

Where 'B' is the number of background tracks in the database.

D. Distributing the RSG to artists and surveying its usage

The RSG program was distributed to over 300 artists across the four genres and they were allowed to test and use excerpts of the song generated by the program for their very own song(sampling). After 3 months, the artists were surveyed on the RSG usage and Sampling and also the overall time is taken for composing their final song.

III. RESULTS

The number of artists sampling the RSG is shown in Table 1 and is categorized based on their genre and experience in music production. Figure 2 shows the percentage of artists with different experience levels using the RSG. According to the survey results, nearly half(144 out of 302) of all the artists involved used the RSG. This is mainly due to the fact though lyrical meaning and text contribute to the songwriting, other factors such as flow, melody, and rhyme scheme play an equal role [4].

TABLE I. NUMBER OF ARTISTS SAMPLING THE RSG

Genre	Experience(in years)	Artists	Artists sampling the RSG
Rock	0-1	36	20
	1-3	21	6
	3-5	26	3
Pop	0-1	31	18
	1-3	22	10
	3-5	10	2
Rap	0-1	41	34
	1-3	37	28
	3-5	34	8
Classical	0-1	21	11
	1-3	16	3
	3-5	7	1

.

The RSG produces an extremely high number of unique flows and rhythms hence artists sample the piece of music which of their choice From Figure 2, it is evident that inexperienced artists(0 to 1 year of experience) found the RSG most useful: 62.32% of all musicians across the genres sampled their music from the RSG at least once in the given period. The artists with 1 to 3 years of experience showed a drop in their RSG usage, with 43.92% of artists sampling the RSG. The same trend followed with the more experienced artists as well. This can be attributed to the age-related differences in susceptibility to irrelevant-speech interference [3]. The time taken to produce new songs with the RSG was also reported by the artists. This was then contrasted with their previous songs composed without RSG and is represented in Figure 3.

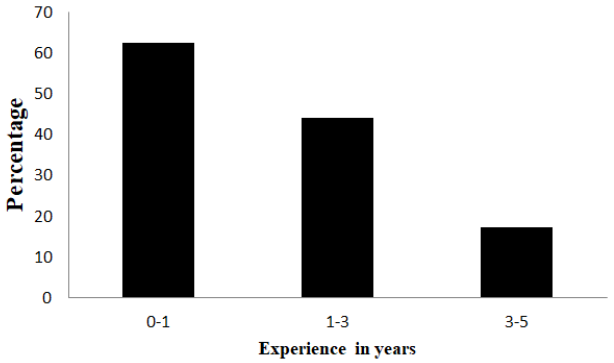

Fig. 2. Artists' experience and their RSG sampling

Figure 3 indicates that the RSG proved beneficial to artists from all four genres in expediting their song composing timeline. The RSG had the best impact when implemented in the RAP genre with over two times the normal song composition speed. The songs were mixed at two separate sampling rates: 44100hertz and 22050 hertz and the time taken the same are shown in Table 2. Based on the analysis of Table 2, it was found that MATLAB was a superior tool to mix the audio tracks for any given frequency as compared to python, with a 77.44% greater speed than python when the song was sampled at 44.1khz and 54.47% greater speed than python when the song was sampled at 22.05khz The same tests were performed with multiple mixing stages and vocal track editing in place, MATLAB outperformed python in terms of speed, and hence MATAB could be used to scale the RSG.

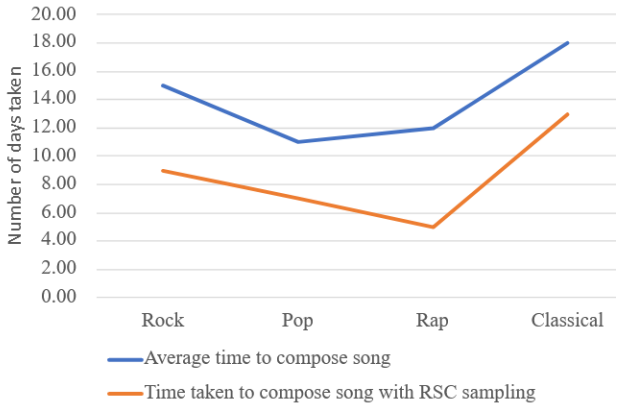

Fig. 3. Time taken to produce a song with and without RSG

TABLE II. TIME TAKEN TO MIX AUDIO TRACKS IN MATLAB AND PYTHON

Song	Sample rate(in hertz)	Time taken in Python	Time taken in MATLAB
1	22050	1.52	1.20
	44100	1.70	1.20
2	22050	1.13	0.69
	44100	1.30	0.69
3	22050	1.26	0.78
	44100	1.44	0.78
4	22050	1.18	0.71
	44100	1.39	0.71

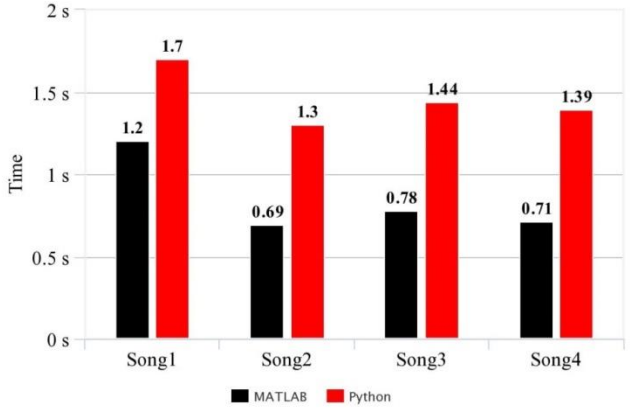

Fig. 4. Time taken for audio mixing in MATLAB and Python for four songs

IV. CONCLUSION

Among the artists, the proportion of them sampling the music from RSG is ~50%. This suggests that lyrics even without any logical meaning can help artists strike an idea when accompanied by some combination of background tracks. The RSG also accelerated music production significantly across all four genres. The Markov chain algorithm can be further investigated and improved to produce better results.

REFERENCES

[1] Ali SO, Peynircioğlu ZF. Songs and emotions are lyrics and melodies equal partners?. Psychology of music. 2006 Oct;34(4):511-34.
[2] Miller C. The differentiated effects of lyrical and non-lyrical music on reading comprehension.352.
[3] Bell R, Buchner A, Mund I. Age-related differences in irrelevant-speech effects. Psychology and aging. 2008 Jun;23(2):377.
[4] Serafine ML, Crowder RG, Repp BH. Integration of melody and text in memory for songs. Cognition. 1984 Apr 1;16(3):285-303.
[5] Nemati, Ali. (2015). Matlab: Matrices.
[6] Giannakopoulos T. pyaudioanalysis: An open-source python library for audio signal analysis. PloS one. 2015 Dec 11;10(12):e0144610.
[7] Viarbitskaya T, Dobrucki A. Audio processing with using Python language science libraries. In2018 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA) 2018 Sep 19 (pp. 350-354). IEEE.
[8] Glover, John C. et al. "Python for audio signal processing." (2011).
[9] Barba P, Maldonado Á, Ortiz M, Plaza E, Vilela L. Voice and Audio Signal Processing using the WSOLA Algorithm MATLAB Software.
[10] Chan KC, Lenard CT, Mills TM. An introduction to Markov chains. InIt's My Maths: Personalised Mathematics Learning, J. Cheeseman, Ed. Proceedings of the 49th Annual Conference of Mathematical Association of Victoria 2012 Dec (pp. 6-7).
[11] Lupyan G, Casasanto D. Meaningless words promote meaningful categorization. Language and Cognition. 2015 Jun;7(2):167-93.
[12] Varnum ME, Krems JA, Morris C, Wormley A, Grossmann I. Why are song lyrics becoming simpler? a time series analysis of lyrical complexity in six decades of American popular music. Plos one. 2021 Jan 13;16(1):e0244576.
[13] Amabile TM, Hadley CN, Kramer SJ. Creativity under the gun. Harvard business review. 2002 Aug 1;80:52-63.
[14] Moorer JA. Music and computer composition. Communications of the ACM. 1972 Feb 1;15(2):104-13.

APPENDIX