



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: XI Month of publication: November 2023

DOI: <https://doi.org/10.22214/ijraset.2023.56534>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Employing LSTMs for Music Remixing

Gourav¹, Aditya Singh², Er. Sandeep Kaur³

^{1,2}Student, ³Assistant Professor, University Institute of Engineering Chandigarh University Mohali, India

Abstract: *The Main Idea was to generate a mixed music from solo musical instrument. For carrying out this task there are various subtask to be carried from data collection to preprocessing the data to make it compatible with the Long Short Term Memory Recurrent Neural Network (LSTM RNNs) model. So, Music mp3 media was collected from melody loops and each one was trimmed down to 30 seconds. Conversion tool was used for segregating the Music into four different mp3 media file i.e. Bass.mp3, Drums.mp3, piano.mp3 and others.mp3. We prepared different models for accomplishing the task to get good results; therefore we preprocessed and transformed the data every time for the new Model architecture. We started with developing the most basic model architecture using Deep neural network (DNNs) and LSTM RNN, and this model architecture was so basic that it did not captures the dependencies within the music, although results was fine but music it generated does not make any sense. Than we tried experimenting by developing different model architectures which generates sensible music and we developed more complex architectures using LSTM RNNs which have the potential to make good results. But there is one disadvantage of it that it takes very long time to get fully trained. In order to reduce the training time and utilizing our resources properly we divided our model into 4 sub-models each for generating music of different musical instruments, These sub-models were trained in parallel on different Google Colaboratory notebook and their best model weights were stored for Reusing them for inference. These all were put into single notebook for predictions and each of these predictions are then combined to make the desired result.*

Keywords: LSTM RNNs, DNNs

I. INTRODUCTION

Till now, Many researches were accomplished in aim to improve the Music generation Model. Whether Music is generated using RNN, LSTM-RNN, GRUs or GANs and this hard effort and time put by many researchers in making the life easier for Music Industry Peoples who always worried about composing quality music. Researches on Artificial Intelligence in the Music Sector drastically reduced the efforts and time needed for composing a single music. One such innovation is added to this pool of Music Generation with the help of our research, which does not focuses generating a music related to single musical instrument but to remix the various musical instrument music to the provided single musical instrument music using LSTM-RNN.

This study attempts to investigate the complex interaction between LSTM neural networks and the craft of remixing music. Because of their effectiveness in broad practical applications, LSTM networks have received a wealth of coverage in scientific journals, technical blogs, and implementation guides [1]. Our goal is to both investigate the technical side of LSTM networks and how they can be used to create remixes that not only pay homage to the original tunes but also add creative inventiveness and fresh components. Our mission is to transform the way music remixes are created by utilizing the power of deep learning [2]. This will enable artists and music lovers to push the frontiers of creativity and create music that is both distinctive and compelling.

II. LITERAUTRE SURVEY

In recent years, there has been a lot of research on using RNNs to generate music in specific genres and styles. For example, in 2020, researchers at Sony AI developed a RNN model that can generate music in the style of Bach chorales. And in 2021, researchers at Google AI developed a RNN model that can generate music in the style of Mozart sonatas [3].

A. Related work

Another important work on music generation using RNNs is the MuseNet model developed by Open AI in 2019. The MuseNet model is a much larger and more complex RNN architecture than the Char-RNN model. MuseNet is trained on a massive dataset of MIDI files, and it can generate music in a variety of styles, including classical, pop, and jazz [4]. In addition to RNNs, other types of deep neural networks have also been used for music generation. For example, in 2017, researchers at the National Tsing Hua University developed a generative adversarial network (GAN) model that can generate music in the style of popular songs [5].

B. Visualizing And Understanding Recurrent Networks

The Music generation using recurrent neural networks is extensive and growing rapidly. One of the earliest works on music generation using RNNs is the Char-RNN model proposed by Andrej Karpathy in 2015 [6]. The Char-RNN model is a simple RNN architecture that is trained on a dataset of text characters [7]. The model can then be used to generate new text, including music lyrics and melodies.

C. Neural Remixer: Learning To Remix Music With Interactive Control

The text discusses the practice of remixing audio recordings by adjusting individual instrument levels and effects. Traditional methods require access to source recordings, limiting creativity. To address this, the authors propose two neural remixing techniques based on Conv-TasNet, allowing direct music remixing. They also employ data augmentation and reconstruction loss, achieving end-to-end separation and remixing [8]. Evaluation on Slakh and MUSDB datasets shows that learning-to-remix surpasses separation methods, especially for minor changes, and provides interactive user controls.

D. Adoption of AI Technology in the Music mixing flows:An Investigation

The field of AI-assisted music production has grown in recent years, offering tools to automate aspects of the process. However, many music producers and engineers are skeptical of these tools, fearing job replacement, doubting AI's ability for subjective tasks, and lacking trust in AI recommendations due to a lack of understanding [9]. To address these concerns, an investigation explores the attitudes and expectations of different user groups employing AI in music mixing workflows. This study involves interviews with professional engineers and questionnaires with both professionals and professional-amateurs. Additionally, internet forums are analyzed to capture a wider range of user sentiments, including beginners and amateurs. The goal is to understand user needs and preferences to develop AI-based mixing tools tailored to each user group's specific requirements.

Overall, the literature on music generation using deep neural networks is very promising. Researchers have developed models that can generate music in a variety of styles, and the quality of the generated music is improving all the time.

III. METHODOLOGY

Different methodology were being tested out for discovering the best model in terms of Fast Training, over fitting, and Quality Music generation [10]. Each of the methodology was under gone through various stages of Data cleaning and processing. Out of them Best methodology was chosen, which gives the state of art results out all applied methodologies by us. So, Following are the different stages we gone through for carrying out this research:

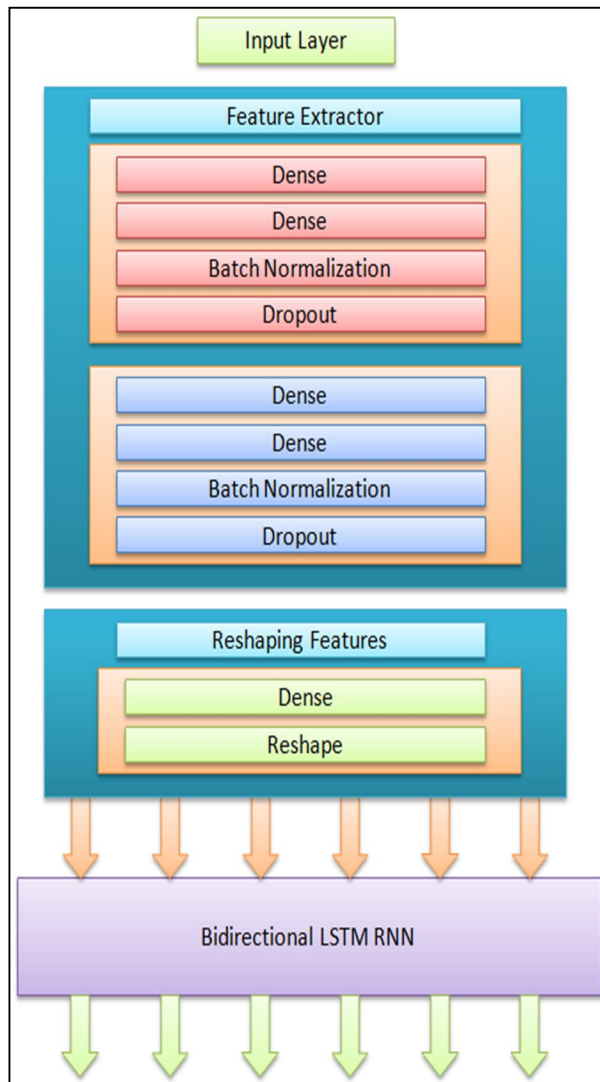
A. Data Collection

For the problem that we are solving, there is no data available on the internet, So, we tried to made our own Dataset containing remix music .mp3 media file and its constituents .mp3 media files. for example, remix.mp3 media file and its constituents are drum.mp3,bass.mp3, piano.mp3 and other.mp3. Although Dataset was not large dataset, It only have 15 .mp3 file and its 4 constituent .mp3 media files, because of lack of computation power. All these medias were collected from melody loops and spitted to into 4 different media files using online conversion tool.

B. Model Architecture Design

Different Model architecture were tested out and discovered some flaws in the architecture which made us to develop the different model architecture for our remix music generation. Every Successive model we built overcomes the limitations of its predecessor or adds new behaviour to this newly built model architecture. Following are the different model architectures we test in order to get best one:

- 1) *One2One Bidirectional LSTM RNNs*: This model architecture consists of feature extraction layers like Dense, Dropout, and Batch Normalizations, and one Bidirectional LSTM RNN layers, with units = 1, timestamps = sample_rate*duration, return sequences = True, Trained with Adam Optimizer having learning rate of 0.0001 [11].Main drawback of this model is the inability to capture the dependency at individual level of musical instrument from previous timestamp. Although model is able to capture the dependency of drum music with piano music and vice versa, but not able to capture the dependency of drum music generated at previous time stamp.



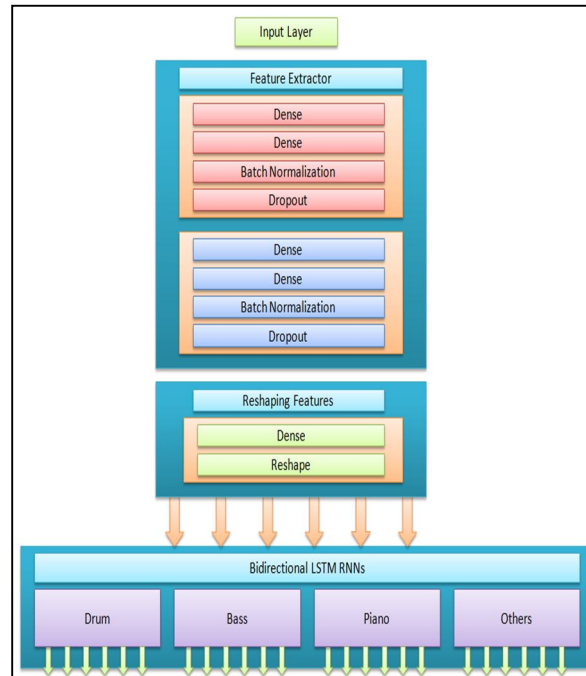
1.One2One Bidirectional LSTM RNN

Due to the dependency of one Musical Instrument with another Musical Instrument One2One Bidirectional LSTM RNN model will able to correlate the Music Generation of one instrument to another without having correlating it with itself.

- 2) *One2Four Bidirectional combined LSTM RNNs*: This one2Four model architecture overcomes the drawbacks of one2one model by introducing 4 Bidirectional LSTM layers [12]. Each for generating drum, piano, bass and others musical instrument music. Configuration of Bidirectional LSTM RNNs is same as one2one model. In this model we also got some flaws that this model able not able to capture the dependency of drum music with piano music and same goes with other musical instruments which one2one model was able to capture. Another problem was its training time; it takes long to train as four models were getting trained simultaneously.

One2Four Bidirectional LSTM RNN combined model is built by taking care providing more control into the relation between Musical instruments itself. It makes used of parallel training of model but due to parallel training resource usage is four times more than the previous one2one Bidirectional LSTM RNN model. Due to the parallel processing, our Google Colaboratory session sometimes got collapsed and we lose all our effort and time made on training the model. To overcome this very much concerning drawback we thought of building new model architecture, which is its next part itself which looks into both dependencies issues and high computation power being used during the training period [13].

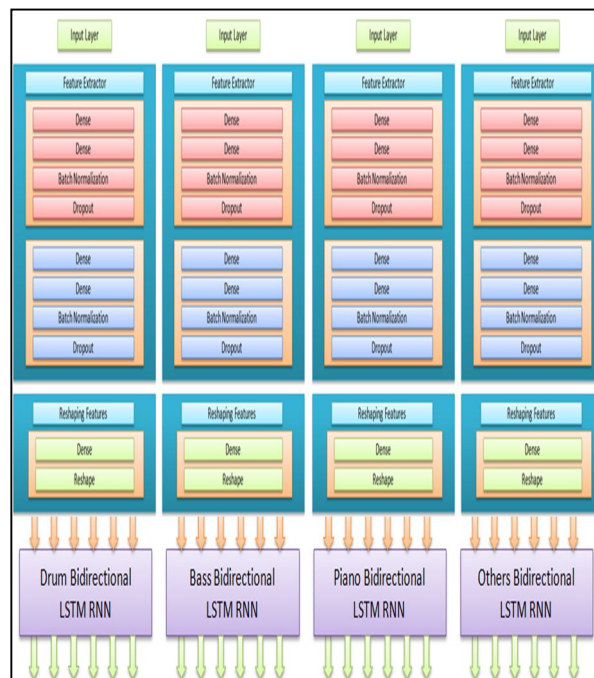
This model architecture is having two abstract layers which are feature extraction layers and reshaping layer which itself consists of Dense, Batch Normalization and Reshape layers with proper configuration each one of this layer for getting good results.



2.One2Four Combined Bidirectional LSTM RNN

There is trade-off exist between the One2One Model and One2Four Bidirectional combined LSTM RNNs, This is because when we try to improve the dependency of single Musical instrument with the other which is choosing One2One model than we loss control on the internal dependency of Musical instrument on itself with previous timestamps.

3) *One2Four Bidirectional splitted LSTM RNNs*: one2Four Bidirectional splitted LSTM RNNs solves the long training time of one2Four Bidirectional combined LSTM RNNs by splitting the one model into 4 models and training is done separately on the same dataset. This reduces the requirement of Higher RAM need at single time. After training each model for drum, piano, bass, and others musical instrument respectively, these model are saved then there results later combined together to produce remix music.



3.One2Four Splitted Bidirectional LSTM RNN

This Model architecture when compared with the other One2One Bidirectional LSTM RNN and One2Four Combined Bidirectional LSTM RNN lacks dependency of one Music Instrument on the another musical instrument, That's why results generated by the model will not be pleasing. But this Overcomes the Drawbacks of one-one drawback of both One2One and One2Four combined Bidirectional LSTM RNN [14].

C. Data Preprocessing

First of all remix and constituents' .mp3 media files was trimmed down to duration of 30 seconds using Librosa. After that Data is splitted into train, validation and test set each containing 9, 9 and 3 files of remix .mp3 files and 4 constituents files [15]. Training of different model architecture is done with different data processing stages in order to find the best model.

D. Flexible Inferencing

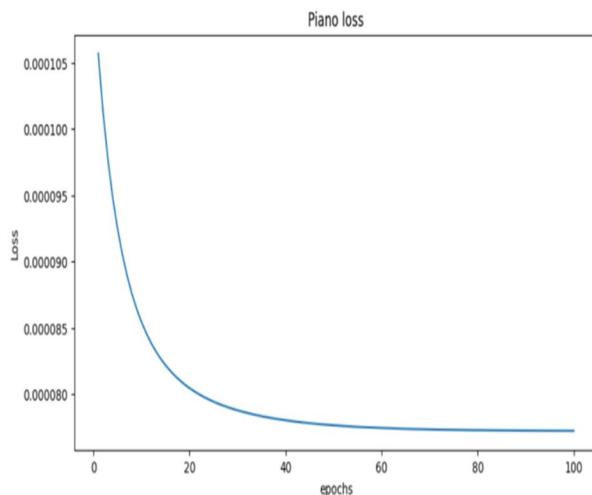
Our Model Architecture developed by us is flexible in nature as it will able to train on any duration of Music Dataset. For the flexible inference our model is trained on one second duration of Music, this is done to provide user flexibility while using our model through any UI because with the training on one second duration of dataset user can request for any duration Music that need to be generated. At the backend many one-duration timestamps will be combined to form the desired duration for Music. Hence this approach of training the model will ensure flexibility to both user which provide better user Experience [16].

IV. RESULTS

After all the evaluation of all three models one2one, one2Four combined, and one2Four splitted Bidirectional LSTM RNN, one model one2Four Splitted Bidirectional and keeping the drawbacks of each of these models one2Four splitted LSTM RNN was selected as the best model to go forward for inference. We will here only show the results for One2Four splitted Bidirectional LSTM RNN model architecture's loss function which we got during model training. As one2Four splitted Bidirectional is the combination of Piano, Drum, Bass and others musical instrument model architectures and their individual training loss function losses are as follows:

A. Loss of Piano Model architecture

This Piano Model architecture of one2one Bidirectional LSTM RNN is trained on only Piano Music Dataset and evaluated using default loss function in keras framework. Below is the loss function graph of training of this model architecture.

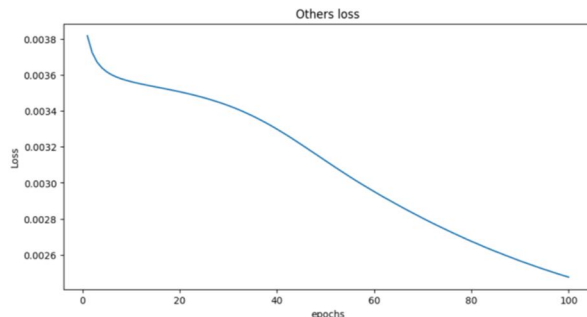


4.Piano one2Four Bidirectional LSTM RNN loss

From this loss function plot we can infer that model reaches the global minima point of loss function at 60th epoch. After 60th epoch model loss reached the saturation point, which is the sign of training need to be terminated.

B. Loss of Others Model Architecture

This Others Model architecture of one2one Bidirectional LSTM RNN is trained on only other Musical instruments Dataset and evaluated using default loss function in keras framework. Below is the loss function graph of training of this model architecture.

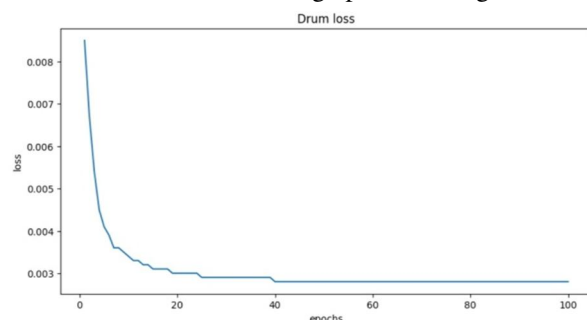


5.Others one2Four Bidirectional LSTM RNN loss

Loss started with 0.00038 loss value and it kept on decreasing in linear fashion on each epoch. From plot it look like we have not reached saturated loss value which means model need to be trained for further more epochs, Here we have limited training to 100 epochs. Also a bit tuning on the learning rate and optimizer we can reach the saturation point of our loss function.

C. Loss of Drum Model Architecture

This Drum Model architecture of one2one Bidirectional LSTM RNN is trained on only Drum Music Dataset and evaluated using default loss function in keras framework. Below is the loss function graph of training of this model architecture.

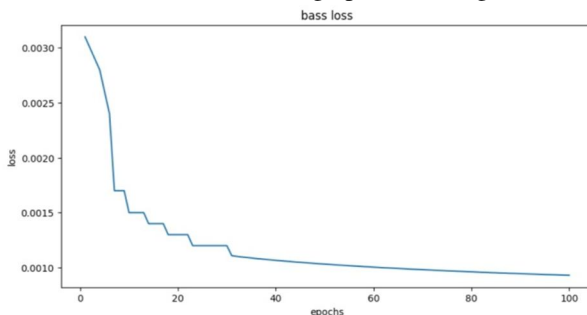


6.Bass one2Four Bidirectional LSTM RNN loss

Here Model loss function started with loss of 0.008 and for first 15 epochs, loss decreases gradually and after 20th epoch loss starts hitting the saturation point. Near 40th epoch loss value reached the saturation point of 0.003. That means model reached the converging point and further training model for more epochs will not affect the loss value and model performance.

D. Loss of Bass Model Architecture

This Bass Model architecture of one2one Bidirectional LSTM RNN is trained on only Bass Music Dataset and evaluated using default loss function in keras framework. Below is the loss function graph of training of this model architecture.



7.Drum one2Four Bidirectional LSTM RNN loss

This model loss is bit jerky in between the epoch number 6 and 30, but after that gradual jerky movement of loss value the model gain stability and started heading toward the saturation point. After few epochs model reached the saturation point leaving behind the jery moment.

V. CONCLUSION

After looking at the several model architectures such as one2one, one2Four combined and one2Four splitted Bidirectional LSTM, our research concluded that one2Four is best model in terms of model training time, and music remixing. But this model also loses in terms of dependency of previous time stamp of one musical instrument on another musical instrument thus not able to generate the good results. In this research some of the drawbacks of Remix Music Generation model have been covered and some are still in pending state like, for problem like getting all the advantages of all three model is still an issue. But with One2Four Splitted Bidirectional model we are able to remix the Music one Musical Instrument. Results generated contain noise in the music thus hide details in the Music which the model has generated. Trying Noise reduction techniques to remove noise have side effects of losing many Musical Notes thus losing some important Musical Notes.

VI. FUTURE SCOPE

The main focus of this project revolves around automating the creation of music using a specialized technology known as a Recurrent Neural Network (RNN) [17]. Importantly, this system is designed to be user-friendly, meaning that even individuals without any prior musical expertise should be able to generate music of reasonable quality with ease.

In essence, the project's primary task is to gather existing music data and use it to train a machine learning model. This model's key mission is to understand the intricate musical patterns that appeal to human sensibilities. Once it comprehends these patterns, the model will have the capability to compose entirely new pieces of music [18]. It's crucial to emphasize that this isn't a process of merely copying existing music; instead, the model is required to analyze these musical patterns and use them as inspiration to create fresh, original compositions.

VII. ACKNOWLEDGMENT

A special thanks to our Supervisor Sandeep Kaur and my colleague Aditya Singh who was always there for Guidance, Encouragement throughout the entire process. Without them, this study would not be timely. They always motivated me to explore more in this field of Music Generation. We are deeply grateful to all of those who helped to make this project a reality, and we hope that our findings will make a meaningful contribution to the field.

REFERENCES

- [1] Alex Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network," Elsevier "Physica D: Nonlinear Phenomena", vol. 404, March 2020.
- [2] Haici Yang, Shivani Firodiya, Nicholas J. Bryan, and Minje Kim, "Neural Remixer: Learning to Remix Music with Interactive Control," eess.as, July 2021.
- [3] Miguel Civit, Javier Civit Masot, Francisco Cuadrado, and Maria J. Escalona, "A systematic review of artificial intelligence-based music generation: Scope, applications, and future trends," Expert Systems with Applications, vol. 209, December 2022.
- [4] Christine Payne. (2019, April) OpenAI. [Online]. openai.com/blog/musenet
- [5] Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang Yang, "MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment," eess.AS, September 2017
- [6] Andrej Karpathy. (2015, May) Andrej Karpathy blog. [Online]. <https://karpathy.github.io/2015/05/21/rnn-effectiveness/>
- [7] Shaily Malik, Gaurav Arora, Anvaya Ahlawat, and Mandeep Payal, "Music Generation Using Deep Learning – Char RNN," Proceedings of the International Conference on Innovative Computing & Communication (ICICC), April 2021.
- [8] Haici Yang, Shivani Firodiya, Nicholas J. Bryan, and Minje Kim, "Neural Remixer: Learning to Remix Music with Interactive Control," eess.AS, July 2021.
- [9] Soumya Sai Vanka, Maryam Safi, Jean-Baptiste Rolland, and György Fazekas, "Adoption of AI Technology in the Music Mixing Workflow: An Investigation," Audio Engineering Society, pp. 13-15, May 2023.
- [10] Tarik A. Rashida, Polla Fattah, and Delan K. Awla, "Using Accuracy Measure for Improving the Training of LSTM with Metaheuristic Algorithms," Procedia Computer Science, vol. 140, October 2020.
- [11] Davide Giordano. (2020, July) towards data science.
- [12] Albert Zeyer, Patrick Doetsch, Paul Voigtlaender, Ralf Schlüter, and Hermann Ney, "A comprehensive study of deep bidirectional LSTM RNNs for acoustic modeling in speech recognition," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2462-2466, 2017.
- [13] Shiliang Sun, Zehui Cao, Han Zhu, and Jing Zhao, "A Survey of Optimization Methods from a Machine Learning Perspective," cs.LG, October 2019.
- [14] Hana Yousuf, Michael Lahzi, Said A. Salloum, and Khaled Shaalan, "A Systematic Review on Sequence to Sequence Neural Network and its Models," International Journal of Electrical and Computer Engineering (IJECE), vol. 11, no. 3, pp. 2315-2326, June 2021
- [15] Darrell Conklin, "Music Generation from Statistical Models," Journal of New Music Research, vol. 45(2), June 2003.
- [16] Alexander Agung Santoso Gunawan, Ananda Phan Iman, and Derwin Suhartono, "Automatic Music Generator Using Recurrent Neural Network," International Journal of Computational Intelligence Systems, vol. 13(1), pp. 645-654, 2020.
- [17] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le, "Sequence to Sequence Learning with Neural Networks," cs.LG, vol. 3, p. 14, December 2014.
- [18] Pedro Ferreira, Ricardo Limongi, and Luiz Paulo Fávero, "Generating Music with Data: Application of Deep Learning Models for Symbolic Music Composition," Applied Sciences, April 2023.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)