

# Pembangkitan Musik Klasik Menggunakan Metode Long-Short Term Memory

Afinzaki Amiral<sup>1</sup>, Muhammad Adrian Surya Saputra<sup>2</sup>, Adam Zufar Majid Suprayogi<sup>3</sup>, Syafira Rosa Amalia<sup>4</sup>, Labib Ahnaf Dhiyaul Khoir<sup>5</sup>, Khafiizh Hastuti<sup>6</sup>, Arry Maulana Syarif<sup>7</sup>

Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

Semarang, Indonesia

<sup>1</sup>111201911857@mhs.dinus.ac.id

<sup>2</sup>111201912248@mhs.dinus.ac.id

<sup>3</sup>111201912115@mhs.dinus.ac.id

<sup>4</sup>111201911822@mhs.dinus.ac.id

<sup>5</sup>111201912109@mhs.dinus.ac.id

<sup>6</sup>afis@dsn.dinus.ac.id

<sup>7</sup>arry.maulana@dsn.dinus.ac.id

**Abstrak**—Penelitian ini bertujuan untuk mengembangkan sistem otomatisasi pembangkitan musik klasik dengan menggunakan Long-Short Term Memory (LSTM), metode yang merupakan bagian dari metode Recurrent Neural Networks (RNN). Pengembangan sistem dilakukan menggunakan Bahasa pemrograman Python. Dataset dalam format MIDI yang dikumpulkan melalui situs [www.piano-midi.de](http://www.piano-midi.de) diolah menggunakan paket Pretty MIDI untuk mengekstrak informasi akord dan notasi dari data yang digunakan sebagai corpus. Selanjutnya, data diekstraksi untuk mendapatkan informasi nada (*pitch*), langkah (*step*) dan durasi (*duration*). Pelatihan jaringan LSTM menggunakan 57.887 notasi yang merupakan hasil ekstraksi data dari 14 file MIDI berisikan komposisi karya Mozart. Hasil pelatihan jaringan LSTM diukur menggunakan metode Mean Squared Error (MSE) dengan akurasi nilai loss dengan nilai loss sebesar 0,43.

**Kata kunci**— algorithmic composition, metode LSTM, pembangkitan musik, data MIDI, Musik Klasik

## I. PENDAHULUAN

Musik adalah seni menyatukan suara dengan menggabungkan kondisi. Ini merupakan hubungan temporal untuk menciptakan komposisi nada dengan kontinuitas dan kesatuan. Musik adalah suara yang dibuat oleh alat musik atau suara manusia. Musik dan suaranya memiliki karakteristik tertentu yang berkaitan dengan kualitas dan kinerjanya. Notasi adalah entitas musik sederhana. Nada-nada yang dihasilkan oleh musik umumnya digambarkan dengan notasi. Terdapat berbagai macam jenis musik di dunia, salah satunya adalah musik klasik. Musik klasik merupakan karya musik yang lahir dari budaya Eropa Timur antara tahun 1750 dan 1825. Berdasarkan eranya, jenis musik klasik dibagi menjadi barok, rococo dan romantis. Dalam sejarahnya terdapat beberapa komposer musik klasik ternama seperti Bach, Mozart dan Haydn. Komposer-komposer ini menghasilkan karya dalam bentuk sonata, simfoni, konserto, kuartet gesek, dan opera. Beberapa instrumen klasik dimainkan secara harmonis untuk menghasilkan nada dan suara yang

indah, seperti piano klasik, gitar klasik, terompet klasik, klarinet, dan biola.

Penelitian ini bertujuan untuk mengembangkan sistem pembangkitan musik klasik berbasis kecerdasan buatan. Pendekatan kecerdasan buatan dapat mengelola urutan notasi-notasi yang memiliki frekuensi yang berbeda dengan menjaga hubungan antar-notasinya. Pendekatan kecerdasan mampu menghasilkan output musik yang secara kualitas dapat diterima oleh manusia. Berbagai metode kecerdasan buatan telah digunakan untuk membangkitkan berbagai jenis musik seperti Sistem berbasis Pengetahuan, algoritme Genetika, Jaringan Saraf Tiruan (JST), termasuk metode Pembelajaran Mendalam. Metode Recurrent Neural Networks (RNN) merupakan bagian dari pembelajaran mendalam yang dapat bekerja secara efektif dalam pencarian solusi untuk permasalahan data dengan tipe time series, termasuk digunakan untuk penyelesaian masalah dalam pembangkitan musik, contohnya penelitian yang dilakukan oleh [1-3]. Hal ini karena beban yang diberikan lebih dekat dan tepat dengan keadaan setiap kata. Artinya, kegiatan serupa akan dilakukan untuk setiap komponen urutan dan hasilnya akan tergantung pada informasi saat ini dan aktivitas sebelumnya. Faktanya adalah bahwa RNN berfokus pada ide-ide informasional dimana kejadian masa lalu atau sekarang ( $t$ ) mempengaruhi kejadian berikutnya ( $t + 1$ ). Salah satu jenis RNN yang digunakan adalah metode Long-Short Term Memory (LSTM). Diluncurkan pada tahun 1997, jaringan LSTM adalah struktur jaringan otak simulasi yang berbeda dari metode RNN. Metode LSTM tidak memiliki masalah dengan kehilangan gradien. Model ini lebih baik daripada RNN sederhana. Perbedaan mendasar pada algoritme ini adalah, metode LSTM dapat mengatur memori pada setiap masukannya dengan menggunakan memory cells dan gate units pada setiap neuron-nya yang berfungsi untuk mengatur memori dalam setiap neuron.

## II. STUDI PUSTAKA

Metode LSTM telah digunakan dalam berbagai penelitian pembangkitan jenis musik, seperti pembangkitan musik pop

[4], pembangkitan musik yang melibatkan kontrol emosi dalam musik [5] dan pembangkitan musik Jazz [6]. Metode LSTM dapat menghasilkan musik secara otomatis tanpa campur tangan manusia. Jaringan LSTM belajar dari kumpulan data dan menganalisisnya untuk membuat kumpulan data murni. Model membutuhkan kemampuan untuk mengingat detail masa lalu dan merekam struktur untuk memprediksi urutan pembelajaran di masa depan. Model perlu melatih dan mentransformasikan urutan asli yang berdekatan dengan urutan sebelumnya untuk sistem pembelajaran. Dalam proses pembangkitan musik, dimungkinkan terjadinya kekeliruan pada mesin. Oleh karena itu dibutuhkan alat atau bantuan yang dapat membantu mesin dalam mengurangi tingkat kekeliruan (*false*) dalam pembangkitannya. Algoritma optimasi Adaptive Moment Estimation (Adam Optimizer) merupakan perluasan untuk Stochastic gradient descent yang telah digunakan sebagai pembelajaran mendalam dalam bidang computer vision dan Natural Language Processing. Algoritme ini dikembangkan dengan memanfaatkan kelebihan dari algoritma Adaptive Gradient (AdaGrad) dan Root Mean Square Propagation (RMSProp). Alih-alih mengadaptasi tingkat pembelajaran parameter berdasarkan rata-rata pertama (mean) seperti dalam RMSProp, Algoritme Adam Optimizer juga menggunakan rata-rata kedua dari gradien (varians uncentered). Algoritme Adam Optimizer yang dikembangkan sebagai dasar untuk pembuatan model pembelajaran mesin dapat digunakan untuk mengolah model data dengan tujuan membangkitkan musik baru. Algoritme Adam Optimizer digunakan oleh [7-8] untuk otomatisasi pembangkitan musik menggunakan metode LSTM. Dalam pelatihan jaringan LSTM, rata-rata kesalahan kuadrat berada di antara nilai aktual dan nilai prediksi. Metode Mean Squared Error (MSE) secara umum digunakan untuk mengecek estimasi jumlah nilai kesalahan dalam prediksi. Nilai MSE yang rendah atau semakin mendekati nol menunjukkan bahwa hasil prediksi sesuai dengan data aktual dan bisa dijadikan untuk perhitungan prediksi di periode mendatang.

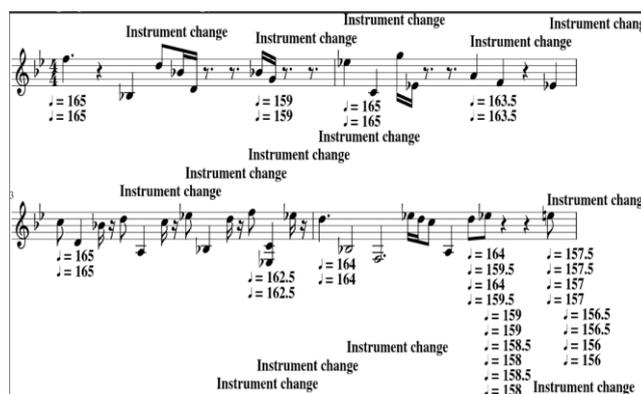
### III. METODE PENELITIAN

#### A. Pengolahan Dataset

Dataset dalam format MIDI (Musical Instrument Digital Interface) telah digunakan dalam -berbagai penelitian pembangkitan musik, seperti penelitian yang dilakukan oleh [9-10]. Dalam penelitian ini, dataset yang digunakan berformat MIDI dan dikumpulkan dari situs [www.piano-midi.de](http://www.piano-midi.de). Situs tersebut memuat berbagai data dari komposer klasik seperti Franz Schubert, Frederic Chopin, Franz Liszt, Johannes Brahms dan yang lainnya. Penelitian ini menggunakan dataset dari komposisi karya Mozart sejumlah 14 file MIDI. Jumlah ini lebih kecil dari yang digunakan oleh [11] yang menggunakan 92 file MIDI, dan lebih besar dari yang digunakan oleh [12] yang menggunakan 12 file MIDI.

Bahasa pemrograman Python digunakan untuk mengembangkan sistem pembangkitan musik klasik. Dataset dalam format MIDI diproses dengan menggunakan paket Pretty MIDI untuk mengekstrak informasi akord dan notasi dari data yang digunakan sebagai corpus. Paket ini berisikan fungsi dan kelas yang memudahkan pengelolaan data MIDI dalam Python,

seperti metode untuk parsing, modifikasi dan analisis file MIDI. Paket Pretty MIDI digunakan dalam berbagai permasalahan otomatisasi pembangkitan musik seperti pembangkitan musik simbolik multi-track [2], pembangkitan musik berdasarkan emosi tertentu [13], pembangkitan musik berbasis Autoencoder yang bervariasi [14] dan pembangkitan musik klasik simbolik [15]. Pemrosesan data dalam penelitian ini juga menggunakan library Pretty MIDI untuk membaca dan menampilkan notasi dalam MIDI. Setelah mendapatkan data notasi, data diekstraksi untuk mendapatkan informasi nada (*pitch*), langkah (*step*) dan durasi (*duration*). Proses ekstraksi menghasilkan 57.887 notasi dari 14 file MIDI Mozart dalam dataset, sedangkan [12] menggunakan 12 data file Midi yang menghasilkan 38.141 notasi. Gambar 1 memperlihatkan hasil visualisasi dari satu file MIDI dari dataset dalam format notasi balok yang menyertakan informasi nada, Langkah dan durasi.



Gambar. 1 Visualisasi MIDI Mozart

Setelah data divisualisasikan, langkah selanjutnya adalah mengekstraksi seluruh notasi yang ada pada dataset yang sudah disiapkan. Pada ekstraksi notasi, tiga variabel digunakan sebagai tolak ukur dalam pengolahan yaitu nada, nama notasi, dan durasi. Gambar 2 memperlihatkan data hasil ekstraksi informasi notasi.

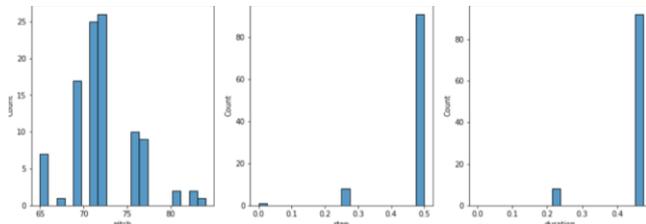
	pitch	start	end	step	duration
0	71	0.0	0.473958	0.0	0.473958
1	69	0.5	0.973958	0.5	0.473958
2	71	1.0	1.473958	0.5	0.473958
3	65	1.5	1.973958	0.5	0.473958
4	71	2.0	2.473958	0.5	0.473958

Gambar. 2 DataFrame dari hasil oleh Datasets

Nada merepresentasikan kualitas persepsi suara sebagai nomor notasi MIDI. Langkah merupakan waktu yang berlalu dari nada sebelumnya atau awal musik. Durasi adalah jumlah waktu dari notasi yang dimainkan dalam hitungan menggunakan unit

pengukuran detik, dan merupakan perbedaan antara waktu awal dan akhir notasi.

Setelah selesai melakukan ekstraksi notasi, langkah selanjutnya adalah mencari distribusi dari notasi yang sudah diolah untuk mengetahui distribusi notasi dari yang paling sering keluar hingga yang tidak pernah atau jarang keluar. Gambar 3 memperlihatkan hasil pencarian distribusi notasi dalam format grafik.



Gambar. 3 Ilustrasi distribusi notasi

### B. Penentuan Data Training

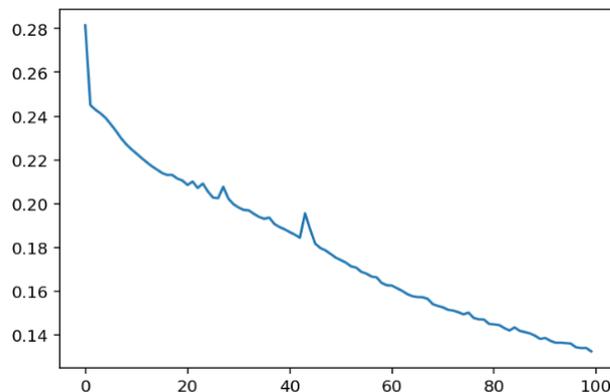
Setelah melakukan eksplorasi data, data *training* ditentukan untuk digunakan dalam pelatihan model LSTM. Data *training* berisi notasi dari file MIDI yang sudah diunduh sebelumnya. Setelah semua data di-parsing, data digunakan untuk melatih kumpulan urutan pada setiap notasi. Setiap contoh urutan berisi catatan yang digunakan sebagai fitur input dan catatan berikutnya sebagai label. Dengan cara ini model dilatih untuk memprediksi nada berikutnya secara berurutan. Pada pengaturan urutan, variable yang diberi-nama panjang\_urutan dideklarasikan untuk menambahkan fitur dan label. Dalam eksperimen, penentuan nilai untuk variable tersebut dilakukan dengan teknik *sampling* untuk mendapatkan nilai terbaik. Hasilnya, nilai 25 ditentukan untuk variabel panjang\_urutan. Nilai tersebut merupakan durasi dari panjang melodi sebesar kurang-lebih 25 detik. Data *training* menggunakan ukuran *batch* sebesar 128 yaitu jumlah notasi dalam format MIDI.

### C. Pelatihan Jaringan LSTM

Setelah melakukan pengolahan dataset, langkah selanjutnya adalah menciptakan dan melakukan *training* pada model LSTM. Arsitektur jaringan LSTM menggunakan tiga output, untuk setiap satu variabel notasi. Implementasi fungsi *custom loss* yang menggunakan nilai *true* dan nilai yang diprediksi berdasarkan parameter yang diatur, dan yang digunakan berdasarkan pada Mean Squared Error yang telah menjadikan model LSTM dapat menghasilkan nilai non-negative, maka model dapat menghasilkan nilai berikutnya.

Setelah menentukan setiap parameter dari masing-masing variabel, pengujian dilakukan dengan menjalankan fungsi *model.evaluate*. Fungsi ini mengevaluasi model yang sudah dilatih menggunakan data validasi dan label yang berkoresponden, serta menghasilkan nilai loss dan matrik untuk model. Pelatihan jaringan LSTM ditentukan dalam 100 kali epoch. Hasil evaluasi menunjukkan tingkat loss sebesar 5,031765937805176, dan tingkat loss pada nada cukup tinggi dibandingkan dengan tingkat loss pada langkah dan durasi, dengan nilai loss nada sebesar 4,848494529724121, nilai loss

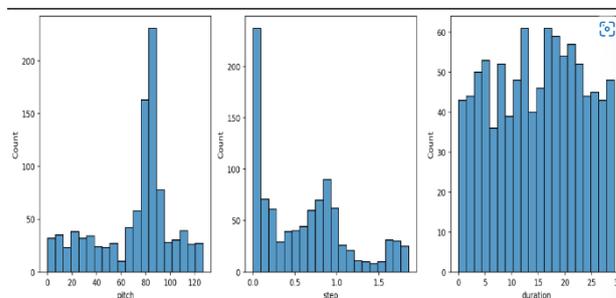
durasi sebesar 0,09059172123670578, dan nilai loss langkah sebesar 0.09268083423376083. Selanjutnya, algoritme Adam Optimizer digunakan untuk meningkatkan performa jaringan LSTM. Hasilnya, terdapat penurunan nilai loss dengan nilai loss sebesar 0,42569711804389954, nilai loss pada nada sebesar 0,09059172123670578, nilai loss pada nada sebesar 4,848494529724121, dan nilai loss pada langkah sebesar 0,09268083423376083. Gambar 4 memperlihatkan grafik histori dari pelatihan jaringan LSTM.



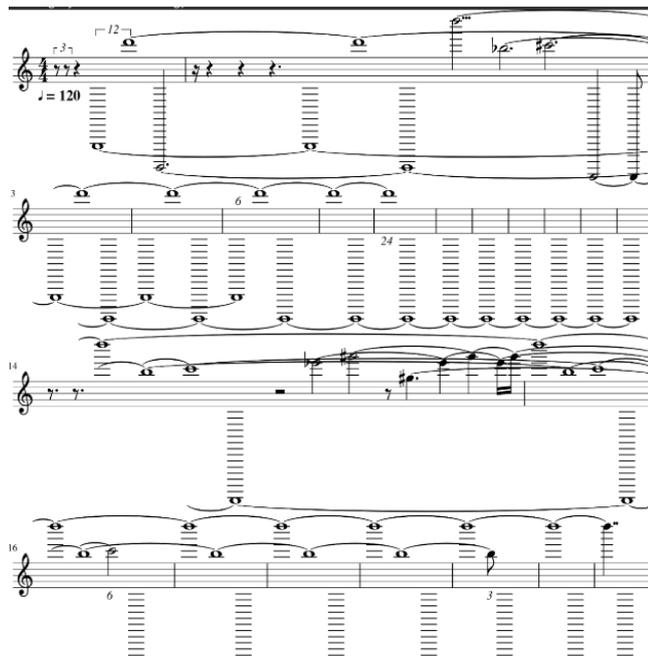
Gambar. 4 Histori pelatihan model

## IV. DISKUSI

Hasil pemodelan LSTM dan pelatihan jaringan LSTM digunakan untuk membangkitkan urutan notasi dengan karakteristik musik Klasik. Fungsi yang dapat membangkitkan setiap notasi pada urutan dirancang dengan mengambil sampel pada nada dari distribusi softmax nada yang dihasilkan oleh model LSTM. Model tidak hanya memilih nada dengan probabilitas tertinggi untuk menghindari pemilihan notasi berdasarkan probabilitas tertinggi yang berdampak pada hasil urutan yang repetitif. Dalam fungsi pembangkitan ini juga dideklarasikan variabel temperature untuk dapat memberikan parameter yang digunakan untuk mengontrol tingkat keacakan dari notasi yang dibangkitkan. Hasil pembangkitan musik menunjukkan distribusi nada, langkah dan durasi telah memenuhi karakteristik dari dataset yang digunakan. Gambar 5 memperlihatkan hasil distribusi nada, langkah dan durasi dari musik Klasik dengan style Mozart yang dibangkitkan, sedangkan Gambar 6 memperlihatkan visualisasi dari musik yang dibangkitkan dalam format notasi balok.



Gambar. 5 Distribusi pada setiap parameter



Gambar.6 Hasil pembangkitan musik Klasik dalam format notasi balok

## V. KESIMPULAN

Pada penelitian ini dapat disimpulkan bahwa komputer sudah Model LSTM yang dikembangkan dengan dataset dalam format file MIDI untuk data *training*, serta implementasi algoritme Adam Optimizer berhasil membangkitkan musik yang memenuhi karakteristik dari dataset yang digunakan. Evaluasi yang dilakukan menggunakan metode MSE dan nilai loss. Pada penelitian mendatang, metode yang diusulkan akan ditingkatkan performanya dengan memodifikasi arsitektur jaringan LSTM, menambah jumlah dataset serta melibatkan pakar musik Klasik untuk mengevaluasi output musik yang dibangkitkan.

## UCAPAN TERIMA KASIH

Artikel ilmiah ini merupakan output dari program Merdeka Belajar oleh mahasiswa yang mengikuti kegiatan Magang Riset dalam pelaksanaan Riset Keilmuan tahun 2021 yang didanai oleh Kementerian Keuangan Republik Indonesia.

## REFERENSI

- [1] X. Liang, J. Wu dan J. Cao, "MIDI-Sandwich2: RNN-based Hierarchical Multi-modal Fusion Generation VAE networks for multi-track symbolic music generation", arXiv:1909.03522 [cs.LG], 2019. Doi: 10.48550/arXiv.1909.03522.
- [2] X. Tan dan M. Antony, "Automated Music Generation for Visual Art through Emotion", dalam Proceedings of the 11th International Conference on Computational Creativity (ICCC'20), 2020, online, September 7 – 11, 2020.
- [3] S. Sajad, S. Dharshika dan M. Meleet, "Music Generation for Novices Using Recurrent Neural Network (RNN)," 2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICES), 2021, pp. 1-6. Doi: 10.1109/ICES52305.2021.9633906.
- [4] S. Mangal, R. Modak dan P. Joshi, "LSTM Based Music Generation System", International Advanced Research Journal in Science, Engineering and Technology, Vol. 6, Issue 5, May 2019. Doi: 10.17148/IARJSET.2019.6508.
- [5] K. Zhao, S. Li, J. Cai, H. Wang dan J. Wang, "An Emotional Symbolic Music Generation System based on LSTM Networks," dalam 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2019, pp. 2039-2043. Doi: 10.1109/ITNEC.2019.8729266.
- [6] J. Wang, X. Wang dan J. Cai, "Jazz Music Generation Based on Grammar and LSTM," 2019 dalam 11th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2019, pp. 115-120. Doi: 10.1109/IHMSC.2019.00035.
- [7] M. Dua, R. Yadav, D. Mamgai dan S. Brodiya, "An Improved RNN-LSTM based Novel Approach for Sheet Music Generation", Procedia Computer Science, Volume 171, 2020, Pages 465-474. Doi: 10.1016/j.procs.2020.04.049.
- [8] T. Jiang, Q. Xiao and X. Yin, "Music Generation Using Bidirectional Recurrent Network," 2019 IEEE 2nd International Conference on Electronics Technology (ICET), 2019, pp. 564-569. Doi: 10.1109/ELTECH.2019.8839399.
- [9] P. Agrawal, S. Kaushik dan S. Banga, "Automated Music Generation using LSTM: Training LSTM based RNN to generate automated music", dalam 5th International Conference on Computing for Sustainable Global Development, 14 - 16 March, 2018.
- [10] S.M. Tony, S. Sasikumar, "Music Generation Using Supervised Learning and LSTM. In: Bansal, R.C.", Agarwal, A., Jadoun, V.K. (eds) Advances in Energy Technology. Lecture Notes in Electrical Engineering, vol 766, 2022. Springer, Singapore. Doi: 10.1007/978-981-16-1476-7\_43.
- [11] A. Ranjan, V.N.J. Behera dan M. Reza, "Using a Bi-directional LSTM Model with Attention Mechanism trained on MIDI Data for Generating Unique Music", arXiv:2011.00773 [cs.SD], 2020. Doi: 10.48550/arXiv.2011.00773.
- [12] A.A.S. Gunawan, A.P. Iman dan D. Suhartono, "Automatic Music Generator Using Recurrent Neural Network", International Journal of Computational Intelligence Systems, Vol. 13(1), 2020, pp. 645-654. Doi: 10.2991/ijcis.d.200519.001;
- [13] K. Zheng, R. Meng, C. Zheng, X. Li, J. Sang, J. Cai, J. Wang, "EmotionBox: a music-element-driven emotional music generation system using Recurrent Neural Network", arXiv:2112.08561 [cs.SD], 2021. Doi: 10.48550/arXiv.2112.08561
- [14] T. Wang, J. Liu, C. Jin, J. Li and S. Ma, "An intelligent music generation based on Variational Autoencoder" 2020 International Conference on Culture-oriented Science & Technology (ICCST), 2020, pp. 394-398. Doi: 10.1109/ICCST50977.2020.00082.
- [15] S. HekmatiAthar, CompoNet: A Novel Hybrid Deep Learning Approach for Symbolic Classical Music Generation, Thesis dari Departemen Ilmu Komputer, North Carolina Agricultural and Technical State University, 2021.