

BAB III

METODOLOGI PENELITIAN

3.1 Objek dan Subjek Penelitian

Objek adalah sesuatu yang menjadi target atau sasaran analisis, tetapi tidak selalu aktif dalam proses penelitian seperti subjek, sedangkan subjek adalah entitas yang menjadi fokus atau pusat dari penelitian. Berikut pembahasan mengenai objek dan subjek dari penelitian ini.

3.1.1 Objek Penelitian

Objek dari penelitian ini yaitu suara latar belakang musik tanpa suara vokal yang diambil dari Spotify karena melihat informasi yang dirilis di laman Spotify.com, saat ini jumlah pengguna berlangganan Spotify telah mencapai 60 juta pada bulan Juli 2017, dengan jumlah pengguna aktif mencapai 140 juta orang yang gemar musik dan tersebar di berbagai belahan dunia. Berdasarkan data visual yang diterbitkan oleh AdWeek, sekitar 72% dari total pengguna Spotify merupakan anggota dari generasi milenial. Generasi milenial telah menjadi segmen pasar terbesar yang menggunakan berbagai platform aplikasi streaming musik seperti Spotify. Contohnya, hingga November 2017, jumlah total pengguna Spotify yang telah mengunduh versi aplikasi mobile Android mencapai 100 juta orang. Perkembangan ini diprediksi akan terus berlanjut seiring dengan meningkatnya penetrasi perangkat telepon pintar yang terus berkembang di Indonesia.

3.1.2 Subjek Penelitian

Subjek dari penelitian ini adalah orang – orang yang menyukai musik khususnya bergenre Pop, karena menurut survei yang dilakukan oleh Catherine Buchholz pada Januari - Desember, 2022 terhadap 21 Negara di dunia yang berusia 18 - 64 tahun ke atas menyatakan bahwa genre yang paling banyak disukai pertama yaitu genre Pop dengan

perolehan data sebanyak 55% responden mengatakan mereka mendengarkannya. Negara yang paling banyak mendengarkan musik pop yaitu Finlandia sebanyak 65%, Swedia sebanyak 61% dan Italia sebanyak 58% [46]. Berdasarkan data dari survei Jakpat yang diperoleh dari <https://dataindonesia.id> pada 7 Desember 2022 terhadap 313 responden, tercatat bahwa sebanyak 26,6% dari partisipan pria menunjukkan minat terhadap genre musik pop. Persentase ini mengalami peningkatan menjadi 30% ketika yang diobservasi adalah responden perempuan. Tak hanya itu, genre musik pop Indonesia juga meraih popularitas yang tinggi di kalangan pria dan wanita di Indonesia. Lebih detail lagi, angka tersebut menunjukkan 25,9% untuk laki-laki dan 24,7% untuk perempuan, yang menunjukkan ketertarikan pada genre musik tersebut [47].

3.2 Alat dan Bahan Penelitian

Alat mengacu pada perangkat fisik atau instrumen yang digunakan dalam penelitian untuk mengumpulkan data, melakukan pengukuran, atau melaksanakan tindakan tertentu, sedangkan bahan merujuk pada bahan - bahan yang digunakan dalam penelitian. Adapun alat dan bahan yang digunakan dalam penelitian ini adalah sebagai berikut.

3.2.1 Alat Penelitian

Terdapat dua perangkat yang digunakan dalam penelitian ini yaitu sebagai berikut.

1. Perangkat keras (*Hardware*)

Perangkat keras yang digunakan dalam penelitian ini ada dua yaitu Laptop Lenovo G40-80 dengan *processor* Intel Core i3, Sistem Operasi Windows dan memori RAM sebesar 16 GB serta Komputer dengan sistem operasi Windows, RAM sebesar 32 GB dan NVIDIA sebagai VGA Cardnya.

2. Perangkat lunak (*Software*)

Perangkat lunak yang digunakan dalam penelitian ini yaitu sebagai berikut.

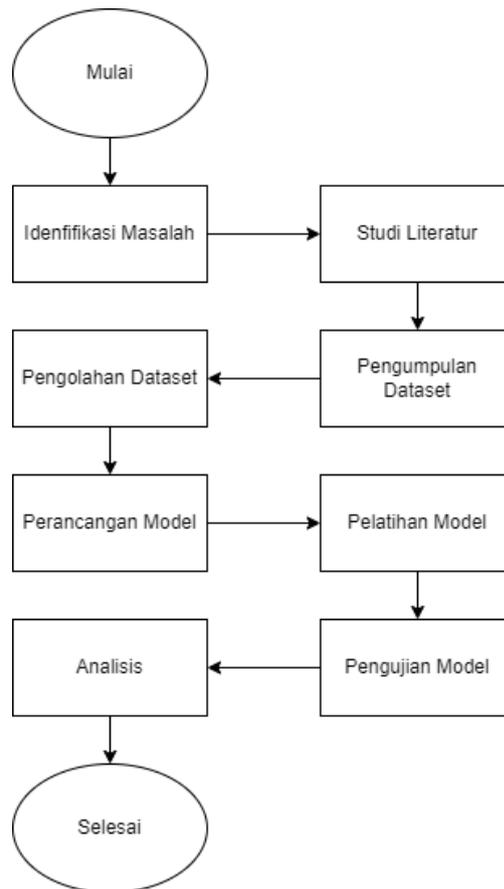
- a. Google Drive digunakan untuk menyimpan dataset, model dan hasil output pada penelitian ini.
- b. Anaconda sebagai *package management system* dan *environment management system* untuk tempat menginstall beberapa package yang diperlukan dan membuat sebuah environment baru dalam menjalankan program.
- c. Google Colab dan Visual Studio Code yang digunakan sebagai *Integrated Development Environment (IDE)* untuk menjalankan program.
- d. Microsoft Word digunakan untuk penulisan naskah tugas akhir.
- e. Microsoft Excel digunakan untuk perhitungan hasil pelatihan.
- f. Windows Media Player digunakan untuk memutar audio yang dijadikan dataset pada penelitian ini.

3.2.2 Bahan Penelitian

Bahan yang digunakan dalam penelitian ini yaitu kumpulan lagu – lagu dengan suara latar belakang tanpa suara vokal berjumlah 100 lagu yang merupakan campuran dari 50 lagu Pop Berbahasa Indonesia dan 50 lagu Pop Berbahasa Inggris yang diunduh dari platform Spotify dengan total durasi sekitar 6 jam.

3.3 Diagram Alir Penelitian

Adapun alur penelitian atau proses penelitian dapat dilihat melalui diagram alir (*flowchart*) sebagai berikut.



Gambar 3. 1 Diagram Alir Penelitian

Proses dari diagram tersebut akan diuraikan dalam beberapa tahap yaitu sebagai berikut.

3.3.1 Identifikasi Masalah

Identifikasi masalah adalah langkah penting dalam tahap perencanaan, karena membantu mengklarifikasi tujuan dan memastikan bahwa upaya yang dilakukan memiliki fokus yang tepat. Pemahaman yang jelas tentang masalah yang akan diatasi juga memungkinkan pengambilan keputusan yang informasi dan solusi yang lebih baik.

Bab ini dimulai dengan membahas latar belakang yang mendorong penelitian pengarang musik otomatis berbasis frekuensi nada dengan menggunakan Recurrent Neural Network (RNN). Latar belakang tersebut mencakup kegunaan musik, berapa banyak orang yang menyukai musik, kesulitan dalam mengarang musik, otomatisasi

dalam mengarang musik, algoritma yang digunakan dalam mengarang musik dan alasan mengapa peneliti menggunakan frekuensi nada dan Recurrent Neural Network (RNN) dalam mengarang musik. Pada bagian rumusan masalah, menjelaskan secara jelas masalah utama yang ingin dipecahkan oleh penelitian ini dan mengapa masalah ini penting dan relevan dalam konteks kecerdasan buatan. Pada bagian pertanyaan penelitian akan dijawab dalam hasil dari bab keempat dalam penelitian ini. Pada bagian batasan masalah, menjelaskan tentang batasan-batasan yang ada penelitian ini. Pada bagian tujuan penelitian, membahas sasaran atau hal yang ingin dicapai dalam penelitian ini. Pada bagian manfaat penelitian, membahas potensi hal yang diperoleh dari penelitian ini

3.3.2 Studi Literatur

Langkah penting dalam penelitian yang melibatkan mengumpulkan, meninjau, dan menganalisis literatur atau sumber-sumber tertulis yang relevan dengan topik penelitian tertentu. Ini membantu peneliti memahami kerangka kerja teoritis yang ada, mengidentifikasi celah pengetahuan, dan mengembangkan landasan teoritis untuk penelitian. Dalam penelitian ini, terdapat lima jurnal yang dijadikan sebagai referensi. Pertama, referensi yang menggunakan Recurrent Neural Network (RNN) dalam studi kasus yang berbeda. Kedua, referensi yang menggunakan metode DCT untuk kompresi data dalam studi kasus yang berbeda. Ketiga, referensi yang menggunakan K-Means untuk mengelompokkan data dalam studi kasus yang berbeda. Keempat dan kelima, menggunakan referensi dalam studi kasus yang sama, tetapi dengan metode yang berbeda yaitu RSCLN Transformer Network dan Generative Adversarial Network (GAN).

3.3.3 Pengumpulan Dataset

Pengumpulan dataset dilakukan dengan menggunakan platform Spotify berupa kumpulan lagu – lagu berjumlah 100 lagu yang terdiri

dari 50 lagu Bahasa Indonesia dan 50 lagu Bahasa Inggris, keduanya merupakan lagu bergenre pop berdurasi sekitar 6 jam. Untuk mengetahui judul lagu yang dijadikan sebagai dataset dalam penelitian ini dapat diakses melalui link berikut.

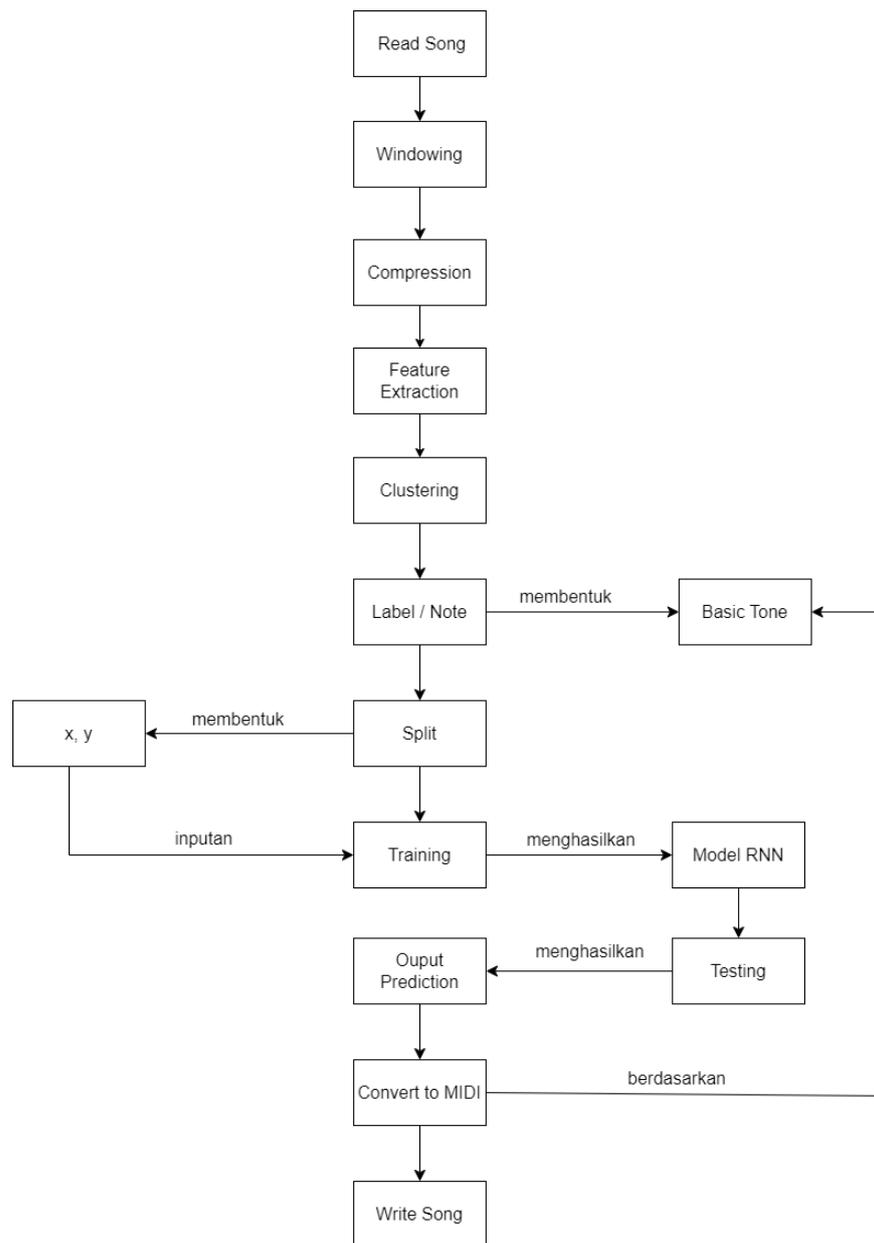
<https://open.spotify.com/playlist/0JVtKk7fe38zLLnnwNLd7D?si=58a2c4f5fb4c4c65>.

Kemudian lagu tersebut dipisahkan suara latar belakang musiknya dengan suara vokalnya dengan menggunakan library dari python yaitu spleeter. Untuk mengetahui dataset yang sudah dipisahkan antara suara vokal dan suara latar belakangnya dalam penelitian ini dapat diakses melalui link berikut.

<https://drive.google.com/drive/folders/1aZQFg8sEskx9lkXgJAAqXWw3UVQRDyMt?usp=sharing>.

3.3.4 Pengolahan Dataset

Proses pengolahan dataset dilakukan untuk mempersiapkan data mentah menjadi format yang lebih sesuai untuk analisis dan eksperimen. Tujuannya adalah untuk memastikan keandalan dan konsistensi data yang akan digunakan dalam penelitian, sehingga hasil yang dihasilkan dari analisis dan eksperimen memiliki dasar yang kuat dan relevan. Langkah – langkah yang dilakukan dalam pengolahan dataset dapat dilihat pada gambar 3.2 berikut.



Gambar 3. 2 Tahap Pengolahan Dataset

Berdasarkan gambar 3.2 tersebut, dapat diuraikan langkah – langkah dalam pengolahan dataset sebagai berikut.

1. Read Song

Suatu proses yang dirancang untuk membaca file audio dalam format WAV. Fungsi ini menggunakan pustaka pydub untuk membaca file WAV. Selama proses membaca, fungsi mengonversi data audio menjadi bentuk array NumPy yang

mewakili sampel-sampel audio. Jika file WAV memiliki dua saluran audio (stereo), data akan diubah menjadi matriks dua dimensi. Opsi `normalized` memungkinkan konversi nilai sampel menjadi `float32` dalam rentang -1 hingga 1.

Proses ini akan menghasilkan `frame rate` (laju frame) dan data audio dalam bentuk array NumPy yang siap untuk diproses lebih lanjut. Fungsi ini berguna dalam membaca data audio dalam format WAV untuk diolah atau dimanipulasi dalam lingkungan Python.

2. Windowing

Fungsi `windowing` digunakan untuk membagi sebuah rangkaian audio menjadi jendela-jendela berukuran tertentu dengan tumpang tindih (`overlap`). Fungsi ini menerima tiga parameter: `musical` (rangkaiannya audio), `window` (ukuran jendela), dan `step` (langkah antara jendela). Selama pemrosesan, fungsi ini mengiterasi melalui rangkaian audio dengan langkah sebesar `step` dan membaginya menjadi potongan-potongan berukuran `window`.

Pemrosesan ini akan menghasilkan array NumPy yang berisi potongan-potongan audio berukuran `window`, yang digunakan nantinya untuk analisis lebih lanjut, yaitu ekstraksi fitur. Pendekatan ini memungkinkan pemrosesan yang lebih efisien dari data audio yang besar dengan meminimalkan duplikasi pemrosesan yang dilakukan pada area tumpang tindih dan untuk menganalisis sinyal secara lebih rinci dengan mengasumsikan bahwa sinyal cukup stabil dalam setiap jendela.

3. Compression

`Compression` adalah salah satu bentuk penggunaan yang penting dari fungsi `DCT` (`Discrete Cosine Transform`), sebuah transformasi matematis yang sering diterapkan dalam analisis sinyal dan pengolahan citra. Dalam konteks penggunaan ini,

DCT digunakan untuk menghitung transformasi DCT dari segmen sinyal audio yang telah di-window dan diolah, yang memungkinkan pengambilan fitur-fitur signifikan dari sinyal tersebut.

Kompresi data yang dilakukan pada penelitian ini menggunakan Discrete Cosine Transform (DCT) yang merupakan kompresi loseless dimana data yang dihasilkan mewakili transformasi domain frekuensi dari sinyal asli. Dalam DCT, informasi yang lebih dominan secara frekuensi diberikan bobot yang lebih besar, sementara informasi frekuensi rendah yang kurang dominan diberikan bobot yang lebih kecil. Hal ini memungkinkan representasi data yang lebih padat dengan mengabaikan komponen frekuensi yang dianggap kurang signifikan, sehingga menghasilkan kompresi yang efisien. DCT sering digunakan dalam berbagai aplikasi seperti kompresi gambar dan audio untuk mengurangi ukuran file tanpa mengorbankan informasi penting secara signifikan. Discrete Cosine Transform (DCT) akan menghasilkan representasi domain frekuensi dari potongan audio yang diberikan. DCT akan mengubah potongan audio menjadi serangkaian koefisien DCT, di mana setiap koefisien mewakili sumbangan dari komponen frekuensi tertentu dalam potongan audio tersebut.

4. Feature Extraction

Setelah proses transformasi Discrete Cosine Transform (DCT) pada potongan kode sebelumnya, langkah berikutnya adalah ekstraksi fitur dari setiap potongan data hasil transformasi. Fungsi ekstraksi fitur ini memiliki tujuan untuk menggambarkan karakteristik penting dari spektrum frekuensi hasil transformasi DCT, yang dapat digunakan dalam proses analisis dan pengolahan lebih lanjut.

Dalam langkah ini, untuk setiap potongan data, dilakukan perhitungan beberapa statistik penting seperti rata-rata amplitudo, deviasi standar amplitudo, dan frekuensi dominan yang diambil sebagai fitur-fitur representatif dari data tersebut. Perhitungan ini akan menghasilkan dalam bentuk suatu matriks fitur, di mana setiap baris mewakili fitur-fitur dari satu potongan data hasil DCT. Matriks fitur ini kemudian digunakan sebagai input untuk langkah pengelompokan menggunakan algoritma K-Means, di mana tiap fitur merepresentasikan potongan data yang telah diwakili oleh karakteristik statistiknya

5. Clustering

Proses ini secara otomatis mengelompokkan fitur-fitur audio yang memiliki karakteristik yang serupa dan memetakan setiap kelompok ke nilai nada dasar yang paling mewakili kelompok tersebut. Proses ini bertujuan untuk menggolongkan potongan data ke dalam klaster-klaster yang memiliki karakteristik yang serupa, memungkinkan model untuk mengenali pola-pola yang mungkin terdapat dalam data audio. Algoritma clustering K-Means digunakan untuk mengelompokkan fitur-fitur dari data audio yang direpresentasikan sebagai nilai-nilai atribut, seperti rata-rata amplitudo, deviasi standar amplitudo, dan frekuensi dominan.

Proses K-Means dimulai dengan membentuk sejumlah klaster yang diinisialisasi secara acak, dan kemudian iteratif mencari pusat klaster yang paling sesuai dengan data. Pusat klaster ini digunakan sebagai representasi klaster dan digunakan untuk mengelompokkan data ke dalam klaster yang sesuai. Hasil pengelompokan ini disimpan dalam variabel `not_nada`, yang merupakan indeks klaster untuk setiap data. Hal ini memungkinkan untuk mengidentifikasi pola-pola karakteristik dalam data audio berdasarkan fitur-fitur yang diambil, dan

klaster-klaster ini dapat mengindikasikan kelompok-kelompok suara yang serupa dalam dataset audio.

6. Label / Note

Membuat nilai-nilai rata-rata dari kelompok-kelompok data yang dihasilkan dari pengelompokan menggunakan algoritma K-Means. Nilai-nilai ini akan merepresentasikan "nada dasar" dari setiap kelompok yang mungkin muncul dalam data audio. Melalui loop, indeks-indeks yang ada dalam kelompok klaster yang dihasilkan sebelumnya oleh model K-Means diiterasi; di setiap iterasi, rata-rata dari data terbagi yang sesuai dengan klaster tertentu dihitung dengan mengambil rata-rata untuk setiap fitur pada data, dan nilai rata-rata ini kemudian ditambahkan ke dalam kamus `nada_dasar` dengan indeks klaster sebagai kunci, menghasilkan kamus yang menyimpan rata-rata fitur dari setiap klaster.

7. Split

Membantu dalam membagi rangkaian data audio menjadi bagian-bagian yang lebih kecil dan sesuai, yang kemudian dapat digunakan sebagai data latih dan data uji dalam proses pelatihan model. Proses ini berfungsi untuk membagi sebuah rangkaian data menjadi pasangan input dan output dengan ukuran jendela yang ditentukan. Ketika memproses data audio atau deret waktu, jendela ini memungkinkan pemisahan data menjadi bagian-bagian yang lebih kecil untuk tujuan analisis atau pelatihan model.

Proses ini menerima dua parameter: `musical` yang merupakan rangkaian data yang akan dibagi, dan `window` yang merupakan ukuran jendela yang digunakan untuk membagi data. Selama pemrosesan, fungsi ini mengiterasi melalui `musical` dengan jarak langkah yang telah ditentukan oleh parameter `step`, dan setiap bagian dengan panjang `window` akan diambil sebagai

input, sedangkan bagian berikutnya akan diambil sebagai output. Hasilnya adalah dua array NumPy, `list_x` yang berisi input dan `list_y` yang berisi output, yang akan digunakan dalam pelatihan model jaringan saraf.

8. Training

Bertujuan untuk mendefinisikan arsitektur model jaringan saraf rekurens (RNN) dengan menggunakan TensorFlow. RNN adalah jenis model yang khususnya baik untuk memproses data sekuen, di mana urutan elemen memiliki arti dan pola temporal penting. Fungsi utama proses ini adalah untuk menghasilkan model yang menerima input berupa urutan data dan menghasilkan output dalam bentuk urutan juga. Fungsi ini dirancang untuk melakukan pemrosesan urutan pada data, memungkinkan model memahami pola dan dependensi temporal dalam urutan tersebut.

Pada tahap ini, dirancang untuk membangun model jaringan saraf rekurens (RNN) dengan lapisan embedding, lapisan GRU (Gated Recurrent Unit), dan lapisan Dense. Pada konstruktor kelas, terdapat tiga lapisan: embedding, gru, dan dense, yang masing-masing bertanggung jawab atas embedding kata, lapisan GRU dengan opsi mengembalikan urutan dan status, serta lapisan Dense untuk output. Metode call kelas ini mengimplementasikan aliran maju (forward pass) melalui model dengan mengambil input `inputs`, nilai status `states` (jika ada), dan opsi lainnya seperti `return_state` dan `training`. Prosesnya mencakup pengubahan input melalui lapisan embedding, menghitung status GRU (jika tidak ada status awal yang diberikan), dan meneruskan hasilnya melalui lapisan GRU dan lapisan Dense. Jika `return_state` diatur sebagai `True`, model mengembalikan output dan status; jika tidak, hanya output yang dikembalikan. Dengan demikian, kelas ini menyediakan struktur

dasar untuk membangun dan menggunakan model RNN dengan lapisan embedding dan GRU dalam TensorFlow.

9. Testing

Bertujuan untuk menghasilkan prediksi langkah-langkah berikutnya dalam rangkaian menggunakan model jaringan saraf rekurens yang telah dimuat sebelumnya. Dalam tahap ini, variabel diisi dengan 30 elemen dari data yang ingin diuji pada indeks ke-100. Selama loop yang berjalan sebanyak 100 iterasi, model yang telah dimuat dipanggil untuk memprediksi langkah-langkah berikutnya berdasarkan 30 elemen terakhir dari input. Logit hasil prediksi diurutkan secara menurun, dan 7 indeks dengan nilai logit tertinggi dipilih sebagai kandidat untuk prediksi berikutnya. Salah satu dari kandidat ini dipilih secara acak sebagai prediksi berikutnya. Hasil prediksi ini kemudian ditambahkan ke input, sehingga langkah-langkah prediksi berikutnya dapat diandalkan pada langkah-langkah sebelumnya. Potongan kode ini mencoba menghasilkan 100 langkah prediksi berikutnya dengan memanfaatkan model yang telah dimuat dan hasil prediksi yang dihasilkan.

10. Convert to MIDI

Tahap ini bertujuan untuk mengubah urutan prediksi langkah-langkah menjadi urutan nilai-nilai nada dasar yang sesuai. Dalam kode ini, variabel midi awalnya berupa daftar kosong. Melalui loop, setiap indeks dalam input (hasil prediksi) digunakan sebagai kunci untuk mengambil nilai-nilai nada dasar yang ada dalam nada_dasar. Nilai-nilai nada dasar ini kemudian ditambahkan ke dalam daftar midi. Setelah selesai mengumpulkan semua nilai-nilai nada dasar, daftar midi diubah menjadi matriks NumPy dengan dua kolom, di mana setiap kolom berisi nilai-nilai nada dasar dari langkah-langkah prediksi. Matriks ini kemudian ditransposisi sehingga baris-baris

menjadi kolom-kolom dan sebaliknya, menghasilkan matriks 2D yang merepresentasikan nilai-nilai nada dasar dari langkah - langkah waktu dalam urutan prediksi.

Proses ini akan menghasilkan representasi musikal dalam bentuk MIDI dari prediksi yang telah dihasilkan sebelumnya menggunakan model jaringan saraf.

11. Write Song

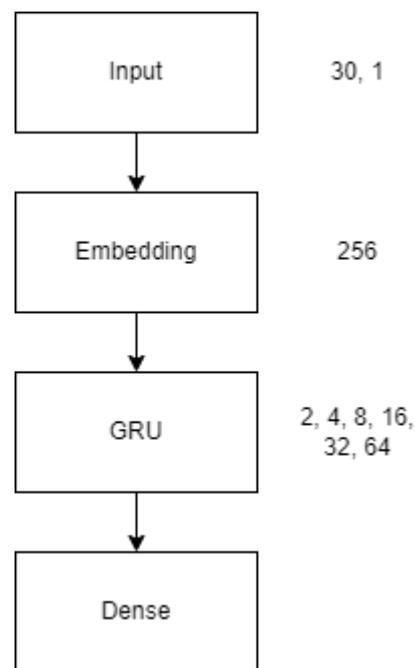
Bertujuan untuk menulis data audio yang diberikan ke dalam format file MP3. Proses ini menerima parameter nama file keluaran, frekuensi sampel, data audio, dan opsi untuk normalisasi. Dalam implementasinya, fungsi ini memeriksa jumlah saluran audio untuk menentukan apakah audio memiliki satu atau dua saluran. Selanjutnya, berdasarkan opsi normalisasi, data audio dikonversi menjadi format integer. Hasil audio yang telah dibuat kemudian diekspor dalam format MP3 dengan bitrate 320 kbps ke dalam file yang ditentukan. Dengan demikian, proses ini menyediakan cara untuk menyimpan data audio dalam format MP3 yang siap digunakan.

3.3.5 Perancangan Model

Model yang digunakan dalam penelitian ini yaitu Recurrent Neural Network (RNN) dengan arsitektur berupa Embedding, GRU, dan Dense. Embedding digunakan untuk mengonversi input yang berupa indeks kata menjadi representasi vektor dengan dimensi sesuai dengan yang diinputkan. GRU digunakan untuk memproses dan memahami informasi sekuensial dari input data. Dense digunakan untuk melakukan *output projection* dari output GRU ke dalam bentuk probabilitas untuk setiap not nada dalam musik. Pada model ini digunakan pengkondisian di mana jika nilai ingin mendapatkan output tambahan dalam bentuk states, dapat mengatur `return_state` menjadi True saat memanggil fungsi. Jika hanya memerlukan output x, dapat

mengatur `return_state` menjadi `False` atau tidak mengaturnya sama sekali.

Pada proses pelatihan ini menggunakan enam model dengan dimensi embedding yang sama yaitu 256 dan GRU Unit yang berbeda mulai dari 2 sampai 64 dengan kelipatan 2. Untuk melihat arsitektur model dapat dilihat pada gambar 4.1 berikut.



Gambar 3. 3 Arsitektur Model

Berdasarkan gambar 4.1 penjelasan langkah – langkah dalam model yaitu sebagai berikut.

1. Pertama, input akan melalui lapisan embedding.
2. Hasil dari lapisan embedding akan digunakan sebagai input untuk lapisan GRU (RNN).
3. Lapisan GRU akan menghasilkan output pada setiap langkah waktu serta status tersembunyi (hidden state) yang akan diteruskan ke langkah berikutnya.

Output dari lapisan GRU akan melewati lapisan Dense untuk menghasilkan prediksi berdasarkan pola yang telah dipelajari oleh model.

3.3.6 Pelatihan Model

Pada tahap pelatihan, model dilatih untuk mendapatkan nilai loss dan validation loss nya. Loss yang digunakan dalam penelitian ini yaitu menggunakan Sparse Categorical Crossentropy yang berfungsi mengukur perbedaan antara distribusi probabilitas yang dihasilkan oleh model dan distribusi probabilitas yang sebenarnya dari data. Ini mengukur seberapa baik model dalam memetakan input ke kelas target yang benar.

3.3.7 Pengujian Model

Pada tahap pengujian, model diuji kekreativitasannya dalam menghasilkan output yang diprediksi dengan mengambil indeks teratas sebagai pilihan nada berikutnya. Pada tahap ini, input yang dimasukkan untuk diprediksi yaitu 30 elemen dari data uji pada indeks ke-100. Kemudian, dilakukan iterasi sebanyak 100 kali untuk menghasilkan prediksi berikutnya. Pada setiap iterasi, Model RNN memprediksi 30 elemen terakhir dari input dan menghasilkan logit prediksi. Matriks logit kemudian diurutkan secara menurun dari yang terbesar ke yang terkecil dengan mengambil indeks nilai logit tertinggi sebagai kandidat untuk prediksi berikutnya. Salah satu dari indeks logit tersebut dipilih dan ditambahkan ke dalam input sebagai elemen berikutnya. Setelah dilakukan 100 hasil prediksi, lalu setiap indeks dalam input digunakan untuk mengambil nilai – nilai nada dasar dan nilai tersebut dimasukkan ke dalam midi. Setelah proses looping selesai, daftar midi yang diambil dari nilai – nilai nada dasar sebelumnya diubah menjadi matriks numpy dengan dua kolom. Setiap kolom mewakili nilai – nilai nada dasar dari prediksi yang dihasilkan. Matriks ini kemudian ditranposisikan sehingga baris – baris menjadi kolom – kolom dan sebaliknya, sehingga hasilnya adalah matriks dua dimensi yang mewakili nada dasar dari setiap Langkah waktu dalam rangkaian prediksi.

3.3.8 Analisis

Setelah dataset dilatih dan diuji, kemudian dianalisis tingkat kegagalannya dan apakah model tersebut underfitting atau overfitting. Kemudian, model manakah yang terbaik dari keseluruhan model yang dibuat. Semakin kecil nilai loss dan validation loss yang didapat, semakin baik model dalam memahami dan memetakan data ke kelas yang benar. Dan model yang baik adalah model yang tidak terdapat underfitting dan overfitting alias bestfitting. Untuk mengetahui model underfitting dan overfitting dapat diketahui dari bias dan variansnya. Untuk menghitung bias dan varians berdasarkan rumus pada bab 2, dapat dihitung dengan cara sebagai berikut.

$$\text{Bias} = \text{Average prediction} - \text{Actual Loss Validation} \quad (3.1)$$

$$\text{Varians} = \text{Mean Squared Deviation} \quad (3.2)$$

dimana Deviasi merupakan pengurangan dari loss training dan rata – rata prediksi.

Hubungan yang terjadi antara bias dan varians yaitu data yang menunjukkan bias yang tinggi namun memiliki varians yang rendah akan menghasilkan keadaan underfitting. Di sisi lain, ketika terdapat bias tinggi dan variance yang tinggi, prediksi menjadi tidak akurat. Ketika bias rendah dan variance tinggi terjadi, situasi overfitting dapat terjadi, di mana model berperforma baik pada data latihan tetapi tidak mampu memprediksi data baru secara efektif. Yang paling diinginkan adalah situasi di mana bias dan variance keduanya rendah [45].