IMPLEMENTASI CNN-LSTM UNTUK MUSIC CAPTIONING

M Ghazali Diarsyah[1], Dhanny Setiawan[2]

Sekolah Tinggi Manajemen Informatika & Komputer LIKMI
Jl. Ir. H. Juanda No.96 Bandung

E-mail : [1]galihdiarsyah@gmail.com,
[2]dhanny2882@gmail.com

ABSTRAK

Musik telah menjadi bagian dari kehidupan manusia sehari-hari. Cakupan musik kini telah meluas ke dalam beberapa sektor industri bahkan, tidak sedikit individu menjadikan musik sebagai kebutuhan hidup. Praktik pemanfaatan *neural network* dalam berbagai subjek penelitian menjadi hal yang lazim tidak terkecuali pada subjek MIR (*music information retrieval)* – bidang penelitian yang secara aktif dijelajahi menggunakan berbagai bidang studi dengan tujuan mengolah informasi musik dan berbagai pengaplikasiannya – arsitektur multimodal *encoder-decoder* yang memanfaatkan algoritma CNN-LSTM merupakan salah satu opsi yang populer. Dengan berbagai pilihan desain arsitektur berdasarkan *modality fusion-early fusion, late fusion dan hybrid fusion,* model dalam penelitian ini menggabungkan pembelajaran data audio dan teks secara bersamaan serta menunjukkan perbandingan dalam hal *music captioning.*

Kata kunci : *CNN, LSTM, Music Information Retrieval, Music Captioning*

ABSTRACT

Music has become an integral part of human life, extending its influence across various industries. For many, music is considered a necessity. With the rise of neural network technology, Music Information Retrieval (MIR) has gained prominence as a multidisciplinary field focused on processing music information and its applications. One popular approach for music captioning is the multimodal encoder-decoder architecture, which utilizes the CNN-LSTM algorithm. In this study, we develop a model that simultaneously learns from audio and text data. We explore different design choices for modality fusion, including early fusion, late fusion, and hybrid fusion, to assess their impact.

Keywords : CNN, LSTM, Music Information Retrieval, Music Captioning

1. PENDAHULUAN

Manusia melalui berbagai macam medium tidak dapat dipisahkan dengan dunia hiburan salah satunya musik. Musik adalah seni membentuk suara dengan menggabungkan vocal dan/atau instrumen menjadi kesatuan. musik sendiri merupakan buah pikiran sehingga elemen vibrasi dalam bentuk frekuensi, amplitudo, dan durasi belum menjadi musik, sampai semua itu di tranformasi secara neurologis dan di interpretasikan melalui otak menjadi pitch timbre, dinamika, dan tempo. [1]

Namun tidak semua manusia mendapatkan anugerah yang sama, beberapa orang dengan gangguan pendengaran tidak mampu memproses musik seperti manusia tanpa gangguan pendengaran, *music captioning* dapat menjadi alternatif bagi orang dengan gangguan pendengaran untuk mendapatkan gambaran tentang musik secara umum. Dengan demikian

orang dengan gangguan pendengaran akan merasa terhubung dengan persepsi yang didapatkan oleh manusia pada umumnya

Belakangan ini pemanfaatan teknik-teknik *deep learning* banyak dipergunakan dalam berbagai subjek penelitian tidak terkecuali MIR (Music Information Retrieval), studi [2] menjelaskan terdapat banyak opsi untuk merepresentasikan data audio dalam mengimplementasikan permasalahan MIR menggunakan *deep learning*, kami memilih Mel-spectrogram dikarenakan keselarasannya dengan sistem pendengaran manusia dalam mempersepsikan frekuensi yang bersifat logaritmik. [3] juga berpendapat bahwa penggunaan mel-spectrogram pada umumnya memberikan performa yang lebih baik dibanding representasi audio lainnya ketika dipergunakan sebagai input pada permasalahan audio yang melibatkan *deep learning*.

Pada persoalan *music tagging* [4] Penerapan algoritma CNN menunjukkan kemampuan yang baik dalam pengolahan data audio. Dan juga memberikan hasil yang lebih baik dibandingkan dengan spectrogram dan MFCC pada dataset yang lebih kecil.

Untuk pekerjaan music captioning [5] khususnya penggunaan arsitektur *multimodal encoder-decoder* yang terdiri dari CNN-LSTM yang kemudian dieksperimentasikan dengan melakukan perbandingan berdasarkan modality fusion yaitu *early fusion* dan *late fusion*. Penambahan arsitektur *hybrid fusion* pada penelitian ini bertujuan sebagai upaya mendapatkan kelebihan dari kedua *modality fusion* yang disebutkan sebelumnya [6], maka penggunannya sebagai metrik perbandingan menjadi salah satu objektif penelitian kami untuk menguji efektivitas dan efisiensi dalam hal *music captioning*. Selain itu juga penggunaan strategi *beam search* decodingnya untuk *music captioning* menjadi acuan kami.

Berdasarkan penelitian terdahulu model kami menggunakan arsitektur *multimodal encoder-decoder* yang memanfaatkan CNN sebagai unit pemroses data audio dan LSTM untuk data teks. Tanpa penggunaan *pre-trained model* untuk pemrosesan data audio, kami melakukan perbandingan dari ketiga *modality fusion* pada dataset yang sama. Serta melakukan proses *hyperparameter tuning* untuk mengoptimalkan performa model.

2. Landasan Teori

2.1. Mel-Spectrogram

Mengacu pada [2], [3] *mel-spectrogram* adalah representasi data audio dalam bentuk 2 dimensi yang dioptimasi untuk mensimulasikan sistem pendengaran manusia, dimana format frekuensi pada axis-y pada *power spectrogram* yang diperoleh melalui penghitungan STFT (*Short Time Fourier Transform)* dikonversi menjadi skala mel.

Proses penghitungan STFT secara matematis dirumuskan dengan persamaan berikut :

$$S(m,k) = \sum_{n=0}^{N-1} x[n]\, w[n-m] e^{-\frac{i2\pi kn}{N}}$$

m: jendela waktu
k : frekuensi
N: *frame-size*
n : sample ke-n
i : unit imaginer
w: fungsi *windowing*

yang kemudian hasil frekuensi k dikonversi menjadi skala mel menggunakan penghitungan :

$$m = 2595 \,.\log(1 + \frac{k}{700})$$

Sehingga hasil kompresi frekuensi data audio bersifat non-linear. Hal tersebut sejalan dengan sistem pendengaran manusia mempersepsikan frekuensi audio berdasarkan *pitch.*

2.2. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah tipe *neural network* yang ditujukan untuk menangani data masukkan dengan struktur grid [7], yang membedakan CNN dengan jenis neural network lainnya yaitu pengoperasian CNN melibatkan setidaknya satu buah layer *convolution*.

Dijelaskan juga bahwa komponen-komponen CNN yang umum digunakan yaitu:

1. *Convolution Layer*
   Setiap layer konvolusi memiliki struktur grid 3 dimensi, yaitu *width, height* dan *depth*. *Depth* pada CNN merujuk pada jumlah *channel* dari layer, seperti nilai dari warna-warna primer contohnya, RGB pada gambar input. Pada *Convolutional Neural Network* parameter-parameternya diatur menjadi unit terstruktur dengan 3 dimensi juga berupa *filter* atau *kernel*. Proses *convolution* pada *convolution layer* terjadi antara *input* atau *feature map* dan *kernel*.
2. *Padding*
   Padding adalah teknik penambahan dimensi pada data input yang bertujuan untuk memelihara informasi pada batas-batas tepi data *input* sehingga tidak merusak informasi data *input*
3. *Pooling*
   Pooling merupakan salah satu metode untuk mengurangi dimensi dari data *input*, penerapan metode *pooling* membantu model untuk menghasilkan *translation invariant*. Hal ini dikarenakan sedikit pergeseran pada input tidak memberikan dampak yang signifikan pada activation map setelah menggunakan pooling. Metode yang umum digunakan pada pooling yaitu *max pooling*, *min pooling* dan *average pooling*.
4. *Fully-Connected Layer*
   Setelah tahap konvolusi dan penggabungan pada *convolutional layer* dan *pooling layer*. Kemudian seluruh vektor ini akan dijadikan sebagai masukan untuk *fully-connected layer*, *fully connected layer* dikenal juga sebagai *dense layer* dimana setiap *neuron* pada layer sebelumnya terhubung dengan setiap *neuron* pada layer berikutnya, struktur dari *fully connected layer* dapat terdiri dari satu atau lebih *hidden layer*.

2.3. Long Short-Term Memory

Long Short-Term Memory (LSTM) merupakan perluasan dari RNN (*recurrent neural network*), LSTM dikembangkan karena desain RNN yang rentan terhadap *vanishing* dan *exploding gradient* [7], Pada LSTM terdapat *hidden vector* tambahan yang sering juga dikenal sebagai *cell state* – state *long-term memory* yang setidaknya membawa sebagian informasi dari state-state pada sel-sel sebelumnya – sehingga operasi *long short-term memory* dapat diformulasikan dengan persamaan :

$$\begin{bmatrix} \bar{\imath} \\ \bar{f} \\ \bar{o} \\ \bar{g} \end{bmatrix} = \begin{pmatrix} sigm \\ sigm \\ sigm \\ tanh \end{pmatrix} W \begin{bmatrix} \bar{x}_t \\ h_{t-1} \end{bmatrix}$$

$$\bar{c}_t = \bar{c}_{t-1} \odot f + \bar{\imath} \odot \bar{g}$$

$$\bar{h}_t = \bar{o} \odot \tanh(\bar{c}_t)$$

Pada setiap sel LSTM secara detail ditunjukkan oleh [8] penggunaan gerbang-gerbang (gates) untuk proses komputasinya, yang dikelompokkan sebagai berikut:

1. *forget gate*
   Forget gate merupakan gerbang yang menentukan besaran informasi dari sel lstm sebelumnya yang akan digunakan untuk menghitung *cell state* dan *hidden state*. Penghitungannya menggunakan persamaan :

$$f_t = \sigma\left(W_f . [h_{t-1}, x_t]\right) + b_f$$

2. *input gate*
   berbeda dengan *forget gate*, *input gate* berfungsi sebagai gerbang penambahan informasi masukkan dari *timestamp* sel yang kemudian menjadi informasi *cell state* dari *timestamp* tersebut. Persamaan yang digunakan yaitu :

$$i_t = \sigma\left(W_i . [h_{t-1}, x_t]\right) + b_i$$

$$\bar{g}_t = tanh\left(W_c . [h_{t-1}, x_t]\right) + b_c$$

3. *output gate*
   Output gate adalah *gate* terakhir yang menentukan *output* dari *cell* dengan melakukan perhitungan nilai *hidden state*. Rumus perhitungannya adalah :

$$\bar{o}_t = \sigma\left(W_o . [h_{t-1}, x_t]\right) + b_o$$

2.4. Encoder - Decoder Model

Secara teori konsep komunikasi [9] istilah *encoding* dan *decoding* dimana *encoding* merupakan proses mentransformasikan informasi menjadi sebuah sandi atau kode-kode tertentu sedangkan *decoding* adalah proses peneriman kode-kode yang kemudian di terjemahkan menjadi sebuah pesan. komunikasi antar model pada arsitektur *encoder-decoder* memiliki interaksi serupa dimana encoder melakukan peran *encoding* dan *decoder* melakukan decoding.

Sedangkan arsitektur *encoder-decoder* untuk *deep learning* [10] merupakan salah satu arsitektur yang umum digunakan pada *sequence-to-sequence* (Seq2Seq) *modeling*. Arsitektur ini utamanya terdiri dari 2 model yaitu *encoder* dan *decoder.*

1. *Encoder*
   Encoder merupakan model penerima sekuens input yang kemudian akan menghasilkan feature vector atau sering juga dikenal sebagai *encoder hidden state* atau *latent representation*. *Feature vector* ini ditujukan untuk menangkap informasi informasi penting dari masukan secara ringkas untuk diproses pada *decoder network*.
2. *Decoder*
   Decoder network menerima *hidden state* dari *encoder* yang kemudian akan digunakan untuk membuat sekuens output, arsitektur ini dapat memanfaatkan berbagai jenis neural network seperti CNN, RNN dan LSTM untuk berbagai macam permasalahan.

2.5. Multimodal Encoder

Multi-modal menunjukkan arsitektur yang memiliki dua atau lebih modalitas, Ketika berbicara mengenai multi-modal pada machine learning maka kata modalitas merujuk pada cara manusia merasakan sesuatu atau juga dikenal dengan *sensory modality* yang mana merepresentasikan saluran komunikasi dan sensasi utama manusia, seperti penglihatan atau sentuhan. Suatu permasalahan atau dataset dikategorikan sebagai *multi-modal* ketika melibatkan beberapa modalitas tersebut [6], juga dijelaskan bahwa terdapat 5 inti permasalahan yang menjadi tantangan dalam mengimplementasikan *multi modal encoder* yakni *Representation*, *Translation*, *Alignment*, *Modality Fusion*, dan *Co-Learning*.

2.6. Categorical Cross Entropy Loss

Categorical cross entropy loss merupakan sebuah *loss function* yang dipergunakan apabila output model berupa distribusi probabilitas dan terdiri dari lebih dari 2 kelas. *Loss function* ini dikenal juga sebagai *log-loss* (*logarithmic loss*) atau *softmax-loss* [7]

Cara menghitung *categorical cross entropy loss* yaitu menegasikan logaritma dari hasil keluaran seperti yang dirumuskan pada persamaan berikut :

$$L = -\log(\tilde{y}_r)$$

2.7. *Adaptive Moment Estimation* (Adam)

Dijelaskan pada [11], [12] bahwa *Adam Optimizer* merupakan metode optimisasi parameter *neural network* yang secara iteratif mengurangi nilai *loss* pada saat *training* yang menggabungkan metode *Stochastic Gradient Descent (SGD)* dengan momentum dan *RMSprop*.

Penghitungan Adam dilakukan dengan menghitung *first moment estimate* atau *momentum* yang diformulasikan sebagai berikut :

$$m^{(t)} = \beta_1 m^{(t-1)} + (1-\beta_1) g^{(t)}$$

Dan *uncentered variance* atau *second moment estimate :*

$$v^{(t)} = \beta_2 v^{(t-1)} + (1-\beta_2)\, g^{(t)} \odot g^{(t)}$$

Penghitungan *gradient* $g^{(t)}$ diperoleh menggunakan turunan parsial dari *loss function* terhadap parameter yang akan di *update*. Adam juga memiliki property *bias correction* terutama pada awal-awal iterasi sehingga rumus *momentum* dan *uncentered variance* yang digunakan yaitu :

$$\hat{m}^{(t)} = \frac{m^{(t)}}{1-\beta_1^t}\,, \hat{v}^{(t)} = \frac{v^{(t)}}{1-\beta_2^t}$$

Hal ini membuat proses konvergensi saat *training* lebih cepat. Pada saat update parameter Adam menggunakan rumus :

$$W_t = W_{t-1} - \eta \frac{\hat{m}^{(t)}}{\sqrt{v^{(t)}} + \varepsilon}$$

Sehingga Adam bersifat adaptif dan *update* parameter yang dilakukan pada setiap parameter bersifat independen, Hal ini membuat Adam cocok untuk data yang *sparse.*

2.8. Beam Search

[13] menjelaskan bahwa *beam search* adalah sebuah algoritma yang banyak dipergunakan pada model NLP dan *speech recognition* sebagai lapisan pengambilan keputusan terakhir untuk menentukan hasil keluaran terbaik, selain itu juga sering digunakan pada model yang memiliki *encoder* dan *decoder* dengan modul LSTM atau *Gated Recurrent Unit* (GRU) di dalamnya. *Beam search* memiliki *hyperparameter beam width* untuk menentukan jumlah alternatif (beams) yang dihasilkan pada saat penghitungan skor setiap langkahnya. Penghitungan skor dalam menentukan posisi urutan beams pada sekuen menggunakan probabilitas bersyarat.

2.9. *Bilingual Evaluation Understudy* (BLEU)

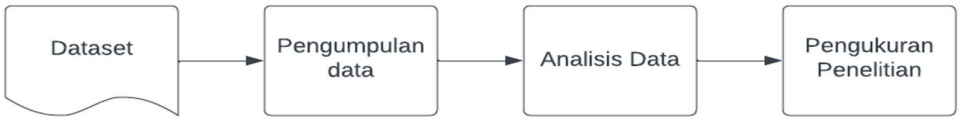
Mengacu pada [14] *Bilingual Evaluation Understudy* (BLEU) merupakan metrik evaluasi untuk membandingkan kandidat kalimat hasil output dengan satu atau lebih kalimat

referensi.

Metrik ini sering digunakan pada kasus-kasus seperti *machine translation, paraphrase* dan *text summarization*. Penghitungan skor BLEU cepat dan tidak memerlukan komputasi yang mahal namun berkorelasi dengan cara manusia mengevaluasi teks.

3. METODE PENELITIAN

Sistem musik captioning menggunakan algoritma CNN–LSTM merupakan sistem pendeskripsian musik secara atmospheric. Dengan memanfaatkan klip musik berdurasi maksimal 20 detik dan caption sebanyak 3-50 token sebagai input. Tahapan-tahapan penelitian dilakukan secara terurut dari dataset hingga metode pengukuran penelitian seperti pada diagram Gambar 1.



Gambar 1. Tahapan-Tahapan Metode Penelitian

a. Dataset
Pada penelitian ini dataset yang digunakan adalah dataset publik MusicCaps yang diambil dari laman web, *https://www.kaggle.com/datasets/googleai/musiccaps* yang terdiri dari 5.521 baris dan 9 kolom,secara detil dijelaskan pada Tabel 1.

Tabel 1. Perincian Dataset MusicCaps

No	Kolom	Keterangan
1	Ytid	identifier video youtube yang akan di segmentasi
2	Start_s	durasi awal klip video
3	End_s	durasi akhir klip video
4	Audioset_positive	Label segmen audio pada dataset Audioset https://research.google.com/audioset/
5	Aspect_list	Daftar aspek yang menggambarkan musik
6	Caption	Deskripsi music
7	Author_id	nomor pengelompokan sampel berdasarkan penulisnya.

b. Pengumpulan Data
Pengumpulan data pada penelitian ini dilakukan dengan pengunduhan klip audio dari platform youtube, mengunakan id yang disediakan pada dataset. Data yang dihasilkan berupa segmen dari audio yang di unduh berdurasi 9 – 20 detik, jumlah data yang berhasil dikumpulkan sebanyak 5.491 dalam format file audio *waveform* (.wav)

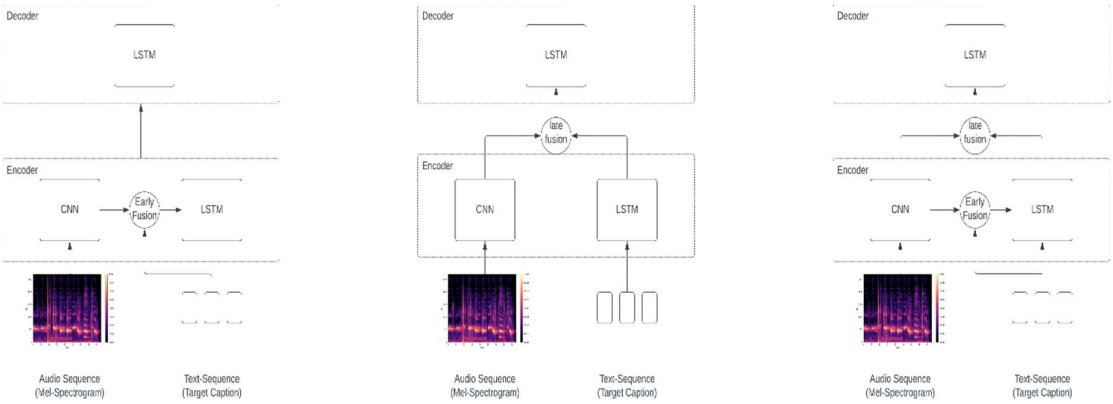
c. Analisis Data
Pada penelitian ini metode analisis data yang paling awal dilakukan yaitu *preprocessing* data, hal ini dilakukan untuk mempermudah proses analisis pada langkah-langkah berikutnya, tahapan *preprocessing* sendiri dibagi menjadi dua yaitu:

1. *Preprocessing* data audio: data audio yang didapat setelah proses pengumpulan data berupa *waveform* yang kemudian diubah menjadi sinyal audio digital. Tujuan dari tahapan ini yaitu mendapatkan *feature* mel-spectrogram dengan *parameter mel-bands* sebanyak 128 dilanjutkan dengan padding data agar membentuk data yang simetris.
2. *Preprocessing* data teks/caption: Beberapa Teknik preprocessing digunakan pada penelitian ini yaitu *tokenization*, *punctuation removal*, *padding*, dan penambahan simbol <startseq> dan <endseq> sebagai simbol awal dan akhir sekuens data text.

   Preprocessing data teks diikuti dengan proses *vectorization* menjadi vector yang berdimensi 300. Proses ini menggunakan teknologi GloVe *word embedding* yang telah di-train menggunakan dataset Wikipedia dan Gigaword 5 GB, bobot dari *embedding matrix* dibiarkan seperti pada saat *pre-trained* ketika proses training berlangsung.[5]

   Setelah melalui preprocessing penerapan *deep learning* digunakan untuk proses analisis data terhadap data-data yang dihasilkan, metode *deep learning* yang digunakan yaitu arsitektur *multimodal encoder-decoder*, metode *training teacher forcing* digunakan dengan cara penambahan simbol ketika *preprocessing* data teks dan mengimplementasikan *data generator* untuk mem-*produce* data pada saat *training.*

   Sebelum melakukan *training*, kami melakukan proses pembagian dataset menjadi dataset training, validasi dan test dengan rasio 70%, 20% dan 10% secara berturut-turut, selain itu kami melakukan proses *hyperparameter tuning* pada ketiga model, hal ini bertujuan untuk mengoptimalkan nilai-nilai *hyperparameter* ketiga model, proses *tuning* menggunakan algoritma *random search* dengan nilai maksimum pecobaan sebanyak 30 kali. Dataset yang digunakan untuk proses *tuning* merupakan sebagian dari dataset validasi. Arsitektur dari ketiga model dibedakan berdasarkan *modality fusion* seperti ditunjukkan pada Gambar 2.



Gambar 2. Overview Arsitektur Berdasarkan Modality Fusion

Seperti pada gambar 2 diilustrasikan arsitektur ketiga model yakni *early fusion, late fusion* dan *hybrid fusion* secara berturut-turut, proses fusion pada ketiga model dilakukan hanya dengan meng-*concatenate* feature dari kedua *modality.* mengacu pada [5] pada bagian multimodal encoder dengan notasi $t = 1 \ldots T$ dimana T adalah jumlah kata pada data caption, dan t menunjukkan kata ke-t pada caption, maka *hidden state* ke-t pada LSTM model encoder dimodelkan dengan persamaan :

$$h_t^{enc} = LSTM(x_t^{enc}, h_{t-1}^{enc}, C_{t-1})$$

x pada persamaan tersebut menunjukkan *input* dan *c* menunjukkan *cell-state* pada LSTM encoder. Pada *early fusion feature-feature* diintegrasikan secara langsung setelah mereka diekstraksi, maka hubungan dan interaksi antar data pada terjadi tahap awal, *feature fusion* sebelum diteruskan pada analysis unit (decoder) untuk pengambilan keputusan, dalam hal ini proses penerjemahan *feature* menjadi sekuens teks atau *caption* sehingga :

$$x_t^{enc} = [a, w_t]$$

Dimana w_t menunjukkan *word embedding matrix* yang diperoleh menggunakan GloVe, dan a merupakan *feature* audio yang diperoleh melalui *fully connected layer* pada network CNN.

Berbeda dengan *early fusion, late fusion* melakukan *feature fusion* terhadap setiap *latent representation* modalitas masing-masing sebelum kemudian *joint representation*-nya dilanjutkan oleh proses decoding pada decoder network. Dengan demikian *input* data audio tidak turut serta mempengaruhi pemodelan sekuens pada LSTM encoder. Maka dari itu *input decoder* x^{dec} adalah :

$$x_t^{dec} = [a, h_{t-1}^{enc}]$$

Pada arsitektur hybrid fusion proses *feature fusion* terjadi pada saat sebelum dan sesudah setiap modalitas melakukan pemrosesan melalui Analysis Unit-nya masing masing (referensi) dalam hal ini CNN dan LSTM. Lalu *joint representation* diteruskan kepada decoder untuk pengambilan keputusan. maka *hidden state early* dan *late* ke-t pada LSTM model encoder dimodelkan dengan persamaan :

$$E_t = LSTM(Ex_t^{enc}, E_{t-1}, C_{t-1})$$
$$L_t = LSTM(w_t, L_{t-1}, C_{t-1})$$

Dimana $Ex_t^{enc} = [a, w_t]$ serta E dan L menotasikan *hidden state* LSTM yang dipergunakkan pada *early* dan *late fusion* masing-masing, proses *fusion* pada bagian *late* menggunakan teknik *concatenation* terhadap L_t sehingga $\overline{L_t} = [a, L_{t-1}]$ dengan demikian *input decoder* x^{dec} adalah :

$$x_t^{dec} = [E_{t-1}, \overline{L}_{t-1}]$$

Untuk bagian *decoder*, LSTM menerima input yang berbeda-beda berdasarkan *modality fusion* yang digunakkan pada tahapan encoding seperti yang telah dijelaskan sebelumnya. LSTM pada decoder diteruskan dengan *Dense* layer dengan neuron sebanyak jumlah kata unik pada data caption, dengan memanfaatkan fungsi aktivasi *softmax* untuk distribusi probabilitas setiap kata.

Ketika proses training, model di-train dengan dataset training dan *categorical cross entropy* sebagai *loss function*. Proses optimisasi menggunakan algoritma *adaptive moment estimation* (Adam). Kami juga menggunakan metode *early stopping* dengan *patience* sebesar 2 dengan *validation loss* sebagai metrik yang dimonitor. *early stopping* diterapkan baik pada saat *training* maupun *hyperparameter tuning*.

Setelah proses *training*, ketiga model menggunakan strategi *beam search decoding* dengan parameter *beam width* sebesar 3 untuk *early fusion* dan 5 pada

kasus lainnya. Pengambilan keputusan akhir berdasarkan *beam* dengan skor tertinggi yang diperoleh pada saat *beam search*.

d. Pengukuran penelitian

Metode Pengukuran penelitian bertujuan untuk mengestimasikan ketepatan hasil caption yang dihasilkan model setelah melalui proses training terhadap data aslinya atau disebut juga *ground-truth*. Metode yang menjadi pilihan kami yaitu metriks BLEU (*Billingual Evaluation Understudy*).

Kemampuan BLEU untuk membandingkan kalimat kandidat dan referensi dengan metode n-gram memberikan kami fleksibilitas untuk memilih besaran unigram, khususnya penelitian ini menggunakan BLEU 1,2 dan 4-gram sebagai metode pengukuran penelitian.

4. HASIL DAN ANALISIS

Ketiga model yang dikembangkan pada penelitian ini yaitu *early fusion, late fusion* dan *hybrid fusion*. Ketiga nilai hyperparameter diperoleh melalui proses *hyperparameter tuning* dengan ruang pencarian seperti diuraikan pada Tabel 2.

Tabel 2. Ruang Pencarian *Hyperparameter Tuning*

Hyperparameter	**Minimum**	**Maksimum**
Jumlah layer *convolution*	1	3
Jumlah filter pada setiap layer *convolution*	32	64
Pooling size	2	3
Rate dropout layer	0.25	0.4

Setelah proses pencarian pemilihan model berdasarkan nilai *hyperparameter* terbaik pada saat *tuning* untuk masing-masing arsitektur. Penentuan model mengacu pada model dengan nilai loss terkecil terhadap dataset validasi. Kemudian masing-masing model dilanjutkan dengan proses *training*.

Proses *training* ketiga model dilakukan dengan memanfaatkan data sebanyak 3853 pasang mel-spectrogram dan caption atau sebesar 70% dataset dan 20% sebagai data *validasi*, 10% sebagai dataset *test* dengan besaran epoch sebesar 50 epoch dengan *batch size* 15, pada saat *training* juga memanfaatkan Teknik *early stopping* dengan *patience*=2. Dengan meminimalisir nilai cross-entropy loss hasil proses training pada kedua model ditunjukkan pada Tabel 3.

Tabel 3. Hasil Proses Training

Model	***Categorical cross entropy loss***	***Validation loss***
Early fusion	1.8580	2.4784
Late fusion	1.6919	2.3418
Hybrid fusion	1.6540	2.3062

Dari Tabel 3 disimpulkan bahwa arsitektur *hybrid fusion* memiliki nilai *loss* lebih kecil dibandingkan dengan kedua model lainnya, meskipun demikian categorical cross entropy loss hanya mengukur seberapa baik algoritma dalam memodelkan data pada saat training, sedangkan untuk mengukur ketepatan caption yang dihasilkan dengan caption *ground-truth* pada penelitian ini digunakan metriks BLEU, maka dari itu proses evaluasi dilakukan dengan ketiga model dengan hasil yang disajikan pada Tabel 4.

Tabel 4. Hasil Proses Evaluasi

Model	Score		
	BLEU-1	**BLEU-2**	**BLEU-4**
Early Fusion	0.722126	0.576129	0.341797
Late Fusion	0.773198	0.617374	0.370171
Hybrid Fusion	0.719942	0.579592	0.351909

Hasil proses evaluasi menunjukkan pada kasus ini model *late fusion* memberikan nilai yang lebih tinggi secara menyeluruh dibanding model lainnya, pada penghitungan score BLEU 2-gram dan 4-gram *hybrid fusion* mengungguli *early fusion*. Hal ini dapat disebabkan karena penghitungan skor BLEU pada 1-gram melakukan akumulasi lebih sering dibanding pada 2 dan 4-gram ketika terjadi perulangan kata. Secara ringkas proses evaluasi ditunjukkan pada Tabel 5.

Tabel 5. Ringkasan Proses Evaluasi

Model	Contoh 1	Contoh 2	Contoh 3
Ground Truth	*This audio contains someone playing a complex groove on tablas along with someone playing a solo on a zitar with the same complex rhythm. This song may be playing in a live concert having a little solo together with instruments.*	*The low quality recording features a repetitive didgeridoo melody. The recording is noisy and in mono, since it was probably recorded with a phone, and it sounds really low in frequency.*	*The low quality, mono recording features a classical song performed by sad violin melody and harpsichord. The recording is noisy and it sounds emotional and passionate.*
Early Fusion	*this is a live performance of a gospel music piece there is a male vocalist singing melodically in the lead the tune is being played by an electric guitar the atmosphere is dreamy*	*the low quality recording features a live performance of a rock song that consists of a flat male vocal talking after which the recording is noisy and in mono*	*the low quality recording features a live performance of a rock song and it consists of passionate male vocal singing over acoustic rhythm guitar melody the recording is noisy and in mono*
Late Fusion	*the low quality recording features a live performance of a traditional song and it consists of a passionate male vocal singing over acoustic rhythm guitar melody it sounds passionate emotional and passionate*	*the low quality recording features a live performance of a rock song and it consists of a passionate male vocal singing over sustained strings melody and sustained strings melody it sounds passionate and passionate*	*the song is an instrumental the song is slow tempo with a piano accompaniment and no other instrumentation the song is emotional and emotional the song is a movie soundtrack*
Hybrid Fusion	*the low quality recording features a live performance of a classical song and it consists of a passionate male vocal singing over acoustic rhythm guitar it sounds emotional and passionate*	*this music is instrumental the tempo is medium with an intense electric guitar harmony with no other instrumentation the audio quality is inferior and the sound of the music is scary and intense*	*the low quality recording features a live performance of a classical song and it consists of a passionate female vocal singing over acoustic guitar melody it sounds passionate and emotional*

Dapat dilihat pada Tabel 5 baik pada hasil yang dicetak oleh model terjadi pengulangan kata terlepas dari jenis *modality fusion* yang digunakan, beberapa penyebabnya antara lain adalah perulangan kata pada dataset ataupun algoritma dan arsitektur yang kurang optimal untuk pengerjaan *music* captioning maka dari itu kami melakukan pengujian pada tahap inferensi dengan menggunakan data yang sebelumnya dilihat oleh ketiga model pada saat *training.* Hasil dari tahap inferensi ditunjukkan pada Tabel 6.

Tabel 6. Hasil Tahap Inferensi

Judul Lagu	Dewa – Separuh Nafas	Peterpan – Bintang di Surga	Project Pop – Keramas
Early Fusion	*the low quality recording features a live performance of a rock song that consists of a passionate male vocal singing over a wide arpeggiated electric guitar melody and shimmering hi hats the recording is a bit noisy and it is a bit noisy*	*the low quality recording features a live performance of a rock song that consists of a passionate male vocal singing over punchy kick and snare hits shimmering hi hats and shimmering hi hats*	*the low quality recording features a rock song that consists of flat male vocal singing over punchy kick and snare hits shimmering hi hats punchy kick and snare hits it sounds energetic and exciting*
Late Fusion	*the low quality recording features a live performance of a rock song that consists of a passionate male vocal singing over shimmering hi hats punchy kick and snare hits shimmering hi hats and groovy bass it sounds energetic and exciting*	*the low quality recording features a live performance of a rock song that consists of a passionate male vocal singing over acoustic rhythm guitar chords and electric guitar melody it sounds energetic and energetic*	*a male singer sings this cool melody with backup singers in vocal harmony the song is medium tempo with a groovy bass line steady drumming rhythm keyboard accompaniment and a groovy bass line the song is emotional and passionate the song is a retro pop song*
Hybrid fusion	*the low quality recording features a live performance of a rock song and it consists of a passionate male vocal singing over punchy kick and snare hits shimmering hi hats and groovy bass it sounds energetic and exciting*	*the low quality recording features a cover of a rock song and it consists of a passionate male vocal singing over acoustic rhythm guitar melody it sounds passionate and emotional*	*the song is an instrumental the song is medium tempo with a groovy bass line steady drumming rhythm keyboard accompaniment and various percussion hits the song is exciting and energetic the song is a modern pop hit*

Hasil pengujian kami menggunakan data yang belum pernah dilihat model sebelumnya ketika training pada Tabel 6 menunjukkan ketiga model mampu melakukan *music captioning* meskipun beberapa fitur dari data musik tidak di-*capture* dengan baik, ketiga model mampu menginterpretasikan input menjadi sebuah *caption.* Bahkan pada kasus tertentu model mampu menggambarkan garis besar dari lagu, khususnya pada hasil pengujian lagu Project Pop – Keramas. Penginterpretasian beberapa model mampu mengklasifikan lagu tersebut sebagai lagu pop yang hit serta menangkap nuansa ceria dan energik dari lagu tersebut. Di sisi lain, perulangan kata dan ketidakcocokan hasil caption dan lagu masih sering terjadi.

5. KESIMPULAN

Dari hasil penelitian ini, dapat disimpulkan bahwa algoritma CNN-LSTM mampu melakukan pekerjaan *music captioning*, dengan ragam arsitektur dan metode-metode yang tersedia pada penelitian ini pemilihan *modality fusion* tidak memberikan dampak yang signifikan, sehingga pemilihan ataupun penambahan metode dan algoritma lainnya sebagai perbandingan pada penelitian selanjutnya perlu dilakukan untuk menemukan model terbaik dalam hal *music captioning*. Selain itu, mengingat kebutuhan *neural network* agar berfungsi dengan optimal memerlukan data dengan jumlah besar, pemilihan dataset yang lebih besar dan pengurangan pengulangan kata kiranya mampu untuk memberikan hasil yang lebih optimal.

DAFTAR PUSTAKA

[1] E. Lararenjana, "Pengertian Musik dan Unsur-unsurnya, Pelajari Lebih Lanjut | merdeka.com." Accessed: May 10, 2023. [Online]. Tersedia: https://www.merdeka.com/jatim/pengertian-musik-dan-unsur-unsurnya-pelajari-lebih-lanjut-kln.html

[2] K. Choi, G. Fazekas, K. Cho, and M. Sandler, "A Tutorial on Deep Learning for Music Information Retrieval," Sep. 2017, [Online]. Tersedia: http://arxiv.org/abs/1709.04396

[3] X. Mei, X. Liu, M. D. Plumbley, and W. Wang, "Automated Audio Captioning: An Overview of Recent Progress and New Challenges," May 2022, doi: 10.1186/s13636-022-00259-2.

[4] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," Jun. 2016, [Online]. Tersedia: http://arxiv.org/abs/1606.00298

[5] I. Manco, E. Benetos, E. Quinton, and G. Fazekas, "MusCaps: Generating Captions for Music Audio," Apr. 2021, doi: 10.1109/IJCNN52387.2021.9533461.

[6] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal Machine Learning: A Survey and Taxonomy," May 2017, [Online]. Tersedia: http://arxiv.org/abs/1705.09406

[7] C. C. Aggarwal, "Neural Networks and Deep Learning : A TextBook," 2018.

[8] Manik Soni, "Understanding architecture of LSTM cell from scratch with code. | by Manik Soni | HackerNoon.com | Medium," Understanding architecture of LSTM cell from scratch with code. Accessed: Apr. 10, 2023. [Online]. Tersedia: https://medium.com/hackernoon/understanding-architecture-of-lstm-cell-from-scratch-with-code-8da40f0b71f4

[9] Anugrah Dwi, "Encoding, Decoding Dalam Komunikasi dan Perbedaanya," https://fisip.umsu.ac.id/encoding-decoding-dalam-komunikasi-dan-perbedaanya/#:~:text=Secara%20sederhana%20encoding%20merupakan%20proses,kode%20untuk%20mengartikan%20sebuah%20pesan.

[10] A. Kumar, "Demystifying Encoder Decoder Architecture & Neural Network - Data Analytics." Accessed: May 02, 2023. [Online]. Tersedia: https://vitalflux.com/encoder-decoder-architecture-neural-network/

[11] Neha Vishwakarma, "What is Adam Optimizer?," https://www.analyticsvidhya.com/blog/2023/09/what-is-adam-optimizer/.

[12] Vitaly Bushaev, "Adam — latest trends in deep learning optimization.," https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c.

[13] M. Payne, "What is Beam Search? Explaining The Beam Search Algorithm | Width.ai." Accessed: May 04, 2023. [Online]. Tersedia: https://www.width.ai/post/what-is-beam-search

[14] Ketan Doshi, "Foundations of NLP Explained — Bleu Score and WER Metrics," https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b.