Hamisfera: Sistem Rekomendasi Progresi *Chord* Berbasis Sentimen Lirik Melalui Studi Komparatif Arsitektur Transformer dan *Mixture of Experts*

Fara Daud Ibra[1], Muhammad Fachrie[2]

Jurusan Informatika, Universitas Teknologi Yogyakarta, Jl. Ring Road Utara, Jombor, Sleman, Daerah Istimewa Yogyakarta 55285, Indonesia[1]
Jurusan Sains Data, Universitas Teknologi Yogyakarta, Jl. Ring Road Utara, Jombor, Sleman, Daerah Istimewa Yogyakarta 55285, Indonesia[2]

Ibragans666@gmail.com[1], muhammad.fachrie@staff.uty.ac.id[2]

Abstract – Songwriting often requires aligning the emotional nuances of lyrics with appropriate musical harmony, yet manually mapping lyrical sentiment to chord progressions is non-trivial. This study identifies the optimal deep-learning architecture for Hamisfera, a two-stage framework that recommends emotionally congruent chord progressions from multilingual lyrics. We conduct a comparative evaluation of Transformer and Mixture of Experts models built on a pre-trained XLM-R backbone, evaluated on a 73,369-entry dataset expanded from 12,000 original entries via augmentation and cleaning. Results are task-dependent, for sentiment classification, the pre-trained MoE attains a slight advantage with an F1-Score of 0.844, whereas for chord generation the pre-trained Transformer performs best overall due to higher generation fluency, with BLEU-4 of 0.608, and high resource efficiency. Accordingly, a hybrid configuration, MoE for classification and Transformer for generation is determined to be the most effective solution.

Keywords – Chord Progression, Mixture of Experts (MoE), Music Generation, Natural Language Processing, Sentiment Analysis, Transformer

Abstrak – Proses penciptaan lagu sering kali memerlukan penyelarasan antara nuansa emosional lirik dengan harmoni musik yang tepat. Namun, penerjemahan sentimen lirik menjadi progresi *chord* secara manual merupakan tantangan yang berarti. Penelitian ini bertujuan untuk mengidentifikasi arsitektur *deep learning* yang optimal bagi sistem Hamisfera, sebuah kerangka dua tahap yang dirancang untuk memberikan rekomendasi progresi *chord* yang relevan secara emosional berdasarkan lirik multibahasa. Metodologi melibatkan studi komparatif antara arsitektur Transformer dan *Mixture of Experts* menggunakan model pralatih XLM-R, yang dievaluasi pada dataset 73.369 entri, diperluas dari 12.000 entri data orisinal melalui augmentasi dan pembersihan. Temuan penelitian menunjukkan hasil yang berbeda per tugas, untuk klasifikasi emosi, arsitektur MoE pralatih menunjukkan keunggulan kecil dengan F1-Score sebesar 0,844. Sebaliknya, untuk generasi *chord*, arsitektur Transformer pralatih unggul secara keseluruhan berkat kefasihan generasi yang lebih baik, dengan BLEU-4 sebesar 0,608, serta efisiensi sumber daya yang lebih tinggi. Oleh karena itu, konfigurasi hibrida MoE untuk klasifikasi dan Transformer untuk generasi ditentukan sebagai solusi paling optimal.

***Kata Kunci* – Analisis Sentimen, Generasi Musik, *Mixture of Experts* (MoE), Pemrosesan Bahasa Alami, Progresi *Chord*, Transformer**

I. PENDAHULUAN

Keterpaduan antara makna lirik dan struktur harmoni menjadi fondasi ekspresivitas musikal. Progresi *chord* berperan sebagai medium utama yang menguatkan narasi emosi [1]. Tantangan muncul saat nuansa yang halus seperti emosi, ketegangan, dan resolusi perlu diterjemahkan ke urutan *chord* yang koheren. Tantangan ini relevan bagi praktik komposisi dan pengembangan sistem komputasional [2],

pISSN: 2252-486X
eISSN: 2548-4710

[3]. *Literatur Music Information Retrieval* menunjukkan bahwa pendekatan *Natural Language Processing* (NLP) pada musik simbolik dapat menjembatani teks dan harmoni melalui representasi sekuens dan mekanisme *attention* untuk konteks jarak jauh [1], [4]. Penelitian ini menargetkan skenario praktis ketika musisi hanya memiliki lirik dan membutuhkan progresi *chord* yang sejalan dengan emosi teks.

Inti masalah terletak pada jarak antara makna lirik dan pemilihan *chord*. Banyak penelitian memakai progresi *chord* sebagai masukan penentu gaya sehingga *chord* diasumsikan sudah tersedia dan tidak diturunkan dari lirik [2], [5], [6]. Akibatnya hasil dapat terdengar musikal tetapi emosi pada teks tidak memandu pemilihan *chord* secara langsung [2]. Pada sisi model, mekanisme *attention* pada Transformer membantu melihat ketergantungan jarak jauh dalam struktur musik namun belum mengikat proses pemilihan *chord* pada representasi semantik lirik [7]. Di tingkat praktik, lagu menuntut pemahaman konteks antarbait agar transisi *chord* tetap masuk akal serta dukungan multibahasa yang stabil yang membutuhkan rancangan data dan model yang tepat [7].

Penelitian ini memperkenalkan Hamisfera sebagai arsitektur dua tahap untuk *sentiment to chord generation*. Tahap pertama mengklasifikasikan emosi pada setiap baris lirik lalu merangkum label mayoritas pada tingkat segmen sehingga tersedia sinyal emosi tunggal yang stabil. Tahap kedua menerima teks segmen beserta label tersebut dan menghasilkan urutan progresi *chord* dalam notasi romawi melalui inferensi *greedy* sehingga keluaran konsisten dan mudah ditransposisi. Seluruh varian memanfaatkan XLM-R pralatih yang kemudian di *fine-tune* pada tugas sasaran guna memperoleh dukungan multibahasa yang kuat [8]. Dataset mencakup 73.369 entri yang dibentuk melalui augmentasi terkontrol dari 12.000 data orisinal dan dikurasi ketat untuk menjaga integritas kualitas. Seluruh eksperimen menggunakan protokol evaluasi yang sama untuk tugas klasifikasi maupun generatif, memastikan perbandingan antarmodel yang adil serta replikasi yang konsisten. Guna mendapatkan arsitektur paling optimal bagi Hamisfera, kami membandingkan dua keluarga arsitektur dengan konfigurasi pelatihan identik. Transformer yang lebih unggul dalam menangkap dependensi jarak jauh [4], serta *Mixture of Experts* (MoE) yang efisien melalui mekanisme *sparse activation* dan skalabilitas parameternya [9], [10].

II. TINJAUAN PUSTAKA

Penelitian ini memusatkan perhatian pada lirik lagu multibahasa dan representasi musik simbolik (progresi *chord*) yang dimodelkan sebagai sekuens token. Objek utama penelitian adalah analisis komparatif performa arsitektur Transformer dan *Mixture of Experts* pada dua tugas spesifik, yaitu klasifikasi emosi lirik dan generasi progresi *chord* terkontrol, di mana keduanya diadaptasi (*fine-tuned*) dari *backbone* lintas bahasa XLM-R [8]. Dalam perspektif teoretis, adaptasi metode NLP ke domain musik simbolik memberikan kerangka kerja yang efektif untuk memetakan hubungan antara teks dan harmoni melalui mekanisme tokenisasi dan pemodelan sekuens [1]. Arsitektur Transformer terbukti efektif dalam menangkap dependensi jarak jauh melalui mekanisme *self-attention* [4], sementara inovasi seperti Museformer menawarkan mekanisme atensi *fine-grained* dan *coarse-grained* untuk mempertahankan struktur repetisi musik dengan cakupan konteks yang lebih luas [7].

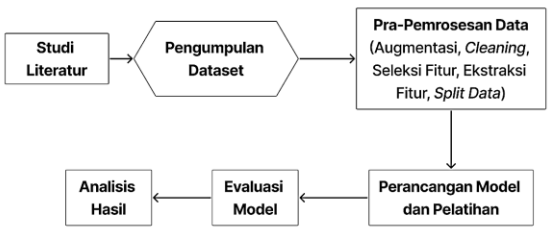
Dalam ranah *controllable music generation*, progresi *chord* lazim digunakan sebagai sinyal kondisi (*conditioning signal*). Model seperti *Chord-Conditioned Melody Transformer* memisahkan *decoder* ritme dan *pitch* agar melodi yang dihasilkan patuh terhadap urutan *chord* [5], sedangkan MusiConGen menambahkan kendali temporal ritme-*chord* baik dari audio rujukan maupun masukan simbolik pengguna [6]. Pada jalur *lyric-to-melody*, sistem TeleMelody menerapkan templat dua tahap (tonalitas, progresi *chord*, ritme, kadensa) untuk meningkatkan kontrol dan efisiensi data [3], sementara ReLyMe menyisipkan aturan teori musik sebagai *soft constraints* untuk meminimalkan disonansi antara lirik dan melodi [2]. Meskipun efektif, pendekatan-pendekatan tersebut umumnya bekerja dengan asumsi bahwa progresi *chord* sudah tersedia sebagai *input*, bukan menurunkannya secara generatif dari teks lirik. Keterbatasan ini memotivasi pengembangan pipeline yang terlebih dahulu mengekstrak

sentimen lirik, kemudian menghasilkan progresi *chord* sebagai fondasi kontrol generatif, sekaligus membandingkan efektivitas keluarga Transformer dan MoE yang skalabel dengan XLM-R untuk skenario multibahasa [8], [9], [10].

Lebih jauh lagi, integrasi aspek multibahasa menghadirkan tantangan spesifik dalam tugas klasifikasi dan generasi yang belum sepenuhnya teratasi oleh pendekatan di atas. Tantangan pertama berkaitan dengan kompleksitas klasifikasi emosi lintas bahasa. Studi terbaru oleh Hong dan Xue [11] menekankan bahwa model konvensional yang dilatih pada dataset monolingual sering gagal menangkap ekspresi emosi dalam konteks linguistik yang beragam, sehingga memerlukan strategi arsitektur multibahasa yang kuat untuk menjembatani kesenjangan semantik antarbahasa (misalnya Inggris dan Mandarin). Tantangan kedua adalah penyelarasan struktural (*alignment*) dalam tahap generasi musik. Karakteristik fonologis yang berbeda antarbahasa (misalnya variasi jumlah suku kata) mempengaruhi pemetaan lirik ke notasi musik. Sheng dkk. dalam sistem SongMASS menyoroti bahwa tanpa mekanisme kendala penyelarasan (*alignment constraint*) yang eksplisit pada level token, model generatif cenderung menghasilkan struktur lagu yang tidak sinkron secara ritmis [12]. Oleh karena itu, penelitian ini memanfaatkan *backbone* XLM-R dan pendekatan hibrida untuk menjawab kedua tantangan representasi tersebut.

III. ANALISA DAN PERANCANGAN SISTEM

Alur penelitian dirancang sistematis untuk memastikan reprodusibilitas dan validitas hasil, seperti divisualisasikan pada Gambar 1.



Gambar 1 Metode Penelitian

Berikut rincian dari alur metode penelitian yang dibuat.

A. Studi Literatur

Berikut adalah landasan teori dari arsitektur dan konsep utama yang digunakan dalam penelitian ini.

1. Transformer

Arsitektur *sequence-to-sequence* berbasis *self-attention* yang menggantikan mekanisme rekurensi, memungkinkan pemodelan dependensi jarak jauh secara paralel dan stabil [4]. Mekanisme intinya adalah *scaled dot-product attention* yang dapat ditinjau pada persamaan 1.

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

2. Mixture of Experts (MoE)

Arsitektur *sparse activation* yang mengganti *feed-forward network* dengan kumpulan *expert* paralel, di mana *router* mengaktifkan subset *expert* per token untuk efisiensi komputasi [9], [10]. Rumus *routing* kombinasi intinya dapat ditinjau pada persamaan 2.

$$MoE(x) = \sum_{i=1}^{n} G(x)_i \cdot E_i(x) \quad (2)$$

3. XLM-Roberta

Encoder Transformer multibahasa, terdiri lebih dari 100 bahasa yang dilatih dengan *masked language modeling* skala besar pada *CommonCrawl* [8]. Representasi lintas bahasa yang dihasilkan sangat kuat untuk *fine-tuning* tugas klasifikasi dan generasi di domain lirik multibahasa.

B. Pengumpulan Dataset

Dataset orisinal disusun dari 12.000 baris lirik lagu multibahasa (Indonesia, Inggris, Jepang) yang dihimpun dari sumber publik daring. Lirik lagu diperoleh dari situs *Genius (genius.com)*, sedangkan progresi *chord* diperoleh dari situs *Ultimate Guitar (ultimate-guitar.com)*. Setiap baris, yang merupakan pasangan lirik dan progresi *chord*, melalui proses kurasi dan pembersihan (deduplikasi, normalisasi simbol, serta pemisahan segmen), kemudian dianotasi secara manual untuk menentukan label emosi yang paling sesuai. Penggunaan data mengikuti ketentuan masing-masing platform dan ditujukan untuk riset akademik dengan atribusi sumber.

pISSN: 2252-486X
eISSN: 2548-4710

C. Prapemrosesan Data

1. Augmentasi Data

Augmentasi data bertujuan untuk menambah variasi lirik lagu multibahasa tanpa mengubah label emosi utama, dengan membatasi perubahan kata seminimal mungkin agar makna tetap terjaga, di mana kami menggunakan kode bahasa IDN untuk Bahasa Indonesia, ENG untuk Bahasa Inggris, dan JPN untuk Bahasa Jepang guna menerapkan pengolahan khusus per bahasa, seperti penyesuaian tokenisasi, normalisasi, keseimbangan data, dan optimasi model MoE melalui pemilihan jalur berdasarkan bahasa agar beban kerja terbagi rata serta evaluasi lebih akurat. Teknik utama augmentasi meliputi *back-translation* (penerjemahan bolak-balik) dan *paraphrasing* berbasis model pralatih seperti *IndoT5-base-paraphrase* untuk IDN (dengan *prompt* sederhana, sampling acak, pembersihan simbol, serta penghapusan duplikat), *T5_Paraphrase_Paws* untuk ENG (dengan filter *cosine similarity* tinggi via *SentenceTransformer* agar makna utuh), serta rute multilingual seperti *M2M100* atau *NLLB-200* untuk JPN (JPN ke ENG/IDN lalu kembali, ditambah *paraphrasing intermediate* menggunakan *T5/IndoT5* secara *batch* untuk efisiensi). *Synonym replacement* berbasis *WordNet* diterapkan dengan batas frekuensi penggantian dan pemeliharaan kapitalisasi, disertai filter semantik serta kolom pelacakan *quality control* seperti *similarity score* dan *augment type*. Hasilnya, jumlah data naik dari 12.000 entri awal menjadi sekitar 120.000 entri, yang meningkatkan ketahanan model XLM-R terhadap perbedaan bahasa dan mengurangi *overfitting* pada kelas emosi yang jarang, sebelum lanjut ke tahap pembersihan data.

2. Cleaning dan Balancing Data

Pembersihan data dimulai dengan normalisasi kode bahasa (ENG, IDN, JPN) dan pembersihan lirik per bahasa melalui *lowercasing*, penghapusan simbol non-esensial, serta penyeragaman spasi. Label emosi dipetakan ke lima kelas utama *(happy, sad, romantic, dark, hopeful*) untuk konsistensi, diikuti eliminasi entri kosong, duplikat, dan *outlier* potensial. Keseimbangan kelas diterapkan secara fleksibel dengan target median distribusi global pada dataset keseluruhan, metode yang digunakan adalah *stratified random oversampling* per strata bahasa–kelas menggunakan fungsi *resample* dari *scikit-learn* dengan *replacement* terbatas maksimum tiga kali per sampel untuk menghindari *overfitting* berlebih, disertai *undersampling* ringan pada strata mayoritas bila diperlukan, lalu data diacak (*shuffle*) untuk mengurangi bias urutan. Prosedur ini menghasilkan sekitar 73.369 entri setelah pembersihan.

3. Ekstraksi Fitur

Ekstraksi fitur dipisahkan untuk lirik dan progresi *chord* agar selaras dengan tujuan tiap tahap model. Untuk lirik, diterapkan tokenisasi *subword SentencePiece* (XLM-R) yang mengubah teks multibahasa menjadi urutan ID token. Pada Model 1 (klasifikasi emosi), lirik diawali token bahasa, misalnya <IDN>, <ENG> dan <JPN> sebagai sinyal eksplisit bahasa. Pada Model 2 (generasi *chord*), lirik didahului header kondisional yang terdiri atas token segmen <SEG_*> (penanda bagian lagu, seperti <SEG_VERSE>, <SEG_CHORUS>), token bahasa (contoh <ENG>), dan token emosi <EMO_*> (penanda kelas emosi target, seperti <EMO_SAD>, <EMO_HAPPY>, <EMO_DARK>). Seluruh token *header* dipisahkan spasi, lalu teks ditokenisasi, *attention mask* membedakan token valid dari *padding*, serta dilakukan *truncation/padding* hingga panjang tetap (MAX_LEN, contoh 128) untuk konsistensi dan efisiensi pemrosesan *batch*.

Untuk progresi *chord*, digunakan rekayasa fitur berbasis aturan (*regex*), pemisah diseragamkan menjadi "-", tiap *chord* diuraikan menjadi derajat Romawi (seperti I, ii, bVI) beserta modifikator kualitas (seperti *maj7, min9, sus4*), lalu dinormalisasi relatif terhadap kunci dasar. Token struktural <BAR> ditambahkan tiap empat *chord* bila informasi birama tidak tersedia, dan <END> di akhir sekuens. Seluruh komponen dipetakan ke kosakata *decoder* tetap (*decoder_vocab.json*) sehingga urutan simbol siap dipelajari secara autoregresif dengan *teacher forcing* tanpa kebocoran informasi masa depan. Deskripsi atribut hasil ekstraksi disajikan pada Tabel 1.

TABEL 1
DESKRIPSI FITUR HASIL EKSTRAKSI

Nama Atribut	Tipe Data	Keterangan
lirik	String	Teks lirik lagu, input utama untuk ekstraksi semantik multibahasa.
Languages	String	Kode bahasa (IDN, ENG, JPN), digunakan sebagai token <LANG_...> untuk *conditioning* bahasa.
Root_label	String	Label emosi utama (contoh: *sad, happy, dark*), dipetakan ke label_id (Model 1) atau token <EMO_...> (Model 2).
Struktur	String	Bagian lagu (contoh: VERSE, CHORUS, INTRO), diubah menjadi token <SEG_...> untuk *conditioning* struktural pada Model 2.
Prog_tokens	String	Urutan *chord* dalam notasi romawi (dipisah spasi), digunakan sebagai target generasi pada Model 2.

4. Pembagian Data

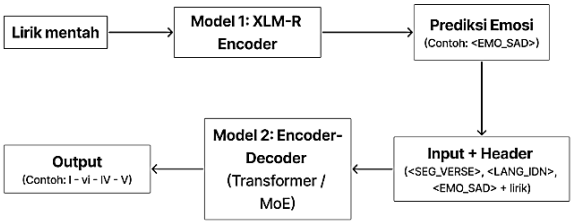
Dataset final dibagi menjadi tiga bagian, 75% untuk pelatihan, 12,5% untuk validasi, dan 12,5% untuk pengujian.

D. Perancangan Model dan Pelatihan

Tahap ini mencakup pelatihan enam varian model utama menggunakan data pelatihan, dengan fokus pada arsitektur Transformer dan MoE yang disesuaikan untuk tugas klasifikasi dan generasi.

1. Arsitektur Global Sistem

Gambar 2 menyajikan arsitektur global sistem yang diusulkan berbasis *pipeline* dua tahap, Model 1 menerima lirik mentah dan menggunakan XLM-Roberta *encoder* untuk memprediksi label emosi (contoh: <EMO_SAD>). Output emosi ini kemudian digabungkan dengan lirik asli dalam bentuk input dengan *header conditioning* yang mencakup token struktur (<SEG_VERSE>), bahasa (<LANG_IDN>), dan emosi (<EMO_SAD>), lalu dimasukkan ke Model 2 berbasis arsitektur *encoder-decoder* (Transformer atau MoE). Model 2 menghasilkan progresi *chord* dalam notasi romawi (contoh: I - vi - IV - V) secara autoregresif melalui *decoder*. Seluruh proses berjalan secara berurutan dan terintegrasi, memastikan harmoni yang dihasilkan selaras dengan emosi dan konteks lirik.



Gambar 2 Arsitektur Global Sistem

2. Varian Model

Sistem ini terdiri atas enam varian model dalam dua kategori, klasifikasi emosi lirik (4 varian) dan generasi progresi *chord* (2 varian). Notasi penamaan, T = Transformer *(encoder-only*), T2 = arsitektur *seq2seq* (*encoder–decoder*) untuk generasi, RAW = pelatihan dari awal, PT = *fine-tuning* dari XLM-R pralatih, dan MoE = integrasi *Mixture of Experts*. Semua model menggunakan XLM-R sebagai *encoder* dan dilatih selama 100 epoch dengan *optimizer* AdamW (*learning rate* = 2e-5, *batch* = 32, *cosine decay*, *label smoothing* = 0,05) untuk menguji pengaruh *fine-tuning* serta efisiensi MoE.

TABEL 2
ARSITEKTUR MODEL KLASIFIKASI

Model	Arsitektur
T-RAW	Transformer encoder dari awal (6 layer), [CLS] + mean pooling → linear classifier
T-PT	XLM-R fine-tune, [CLS] + mean pooling → MLP head
MoE-RAW	T-RAW + *hard routing* MoE (*router* memilih 1 dari 3 expert berbasis bahasa).
MoE-PT	T-PT + MoE Adapter 4 layer (top-k=2, 8 expert, gating bahasa-aware)

Tabel 2 merangkum arsitektur Model 1 untuk klasifikasi emosi lirik dengan variasi *backbone* dan MoE. Varian ini membandingkan pelatihan dari awal versus *fine-tuning*, sekaligus menguji efisiensi *routing* MoE berbasis bahasa. [CLS] adalah vektor token klasifikasi dari *encoder*, *mean pooling* menghitung rata-rata *embedding* token *non-padding*.

pISSN: 2252-486X
eISSN: 2548-4710

TABEL 3
ARSITEKTUR MODEL GENERASI

Model	Arsitektur
T2-PT	T-PT + MoE Adapter 4 layer (top-k=2, 8 expert, gating bahasa-aware)
MoE2-PT	T2-PT + MoEFFN di decoder (top-k=2, 8 expert, load-balance loss)

Seperti terlihat pada Tabel 2, Model 2 menghasilkan urutan chord secara autoregresif menggunakan decoder Transformer dengan conditioning dari emosi dan struktur. Varian MoE menggantikan feed-forward biasa dengan MoEFFN untuk meningkatkan kapasitas dan efisiensi inferensi. Model generasi menggunakan teacher forcing saat pelatihan dan greedy decoding saat inferensi. Evaluasi menggunakan Exact-Match, BLEU-4, ROUGE-L, Edit Distance, Bar-Match, Set F1. Checkpoint terbaik dipilih berdasarkan validasi.

E. Evaluasi Model

Dataset akhir sebanyak 73.369 entri dibagi secara *stratified split* berdasarkan label emosi dan bahasa menjadi 75% data pelatihan (55.027 entri), 12,5% data validasi (9.171 entri), dan 12,5% data pengujian (9.171 entri), di mana set validasi dan pengujian memiliki distribusi label emosi yang seimbang (1.834 entri per kelas) untuk evaluasi yang adil, sedangkan set pelatihan mencerminkan distribusi asli setelah *oversampling*. *checkpoint* adalah catatan lengkap dari seluruh bobot model dan status optimizer yang disimpan setiap 1 epoch selama pelatihan, dengan *checkpoint* terbaik dipilih berdasarkan performa tertinggi pada set validasi (*F1-Score macro* untuk klasifikasi, *BLEU-4* untuk generasi), kemudian kinerjanya dievaluasi secara kuantitatif pada set pengujian menggunakan metrik standar yang diterapkan konsisten di seluruh varian model untuk mengukur akurasi klasifikasi emosi lirik dan kesesuaian struktur sekuens generatif. Metrik evaluasi yang digunakan adalah sebagai berikut.

1. F1-Score (macro)

metrik evaluasi klasifikasi berupa rata-rata harmonis *precision dan recall*, yang memberi gambaran kinerja lebih seimbang terutama saat terjadi ketidakseimbangan kelas [13]. Notasi: TP menunjukkan jumlah prediksi benar pada kelas ke-i; FP adalah jumlah prediksi salah pada kelas ke-i; FN menyatakan data kelas ke-i yang tidak terdeteksi; dan K merupakan jumlah total kelas.

$$Precision = \frac{TP}{TP+FP}\ ;\ Recall = \frac{TP}{TP+FN} \tag{3}$$

$$F1_{\text{macro}} = \frac{1}{K}\sum_{i=1}^{K} 2\frac{Precision \cdot Recall}{Precision+Recall} \tag{4}$$

2. Cross-Entropy Loss

Mengukur seberapa besar perbedaan antara distribusi prediksi model dan label sebenarnya, semakin kecil nilainya semakin baik prediksi model [14]. Notasi: N=jumlah sampel, C= jumlah kelas, $y_{i,j}$=label *one-hot* (1 untuk kelas benar), $p_{i,j}$= probabilitas yang diprediksi untuk kelas j pada sampel i(logaritma natural).

$$L_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C}(y_{i,j}\cdot\log(p_{i,j})) \tag{5}$$

3. Perplexity (PPL)

Mengukur seberapa bingung model bahasa dalam memprediksi token berikutnya, dengan nilai lebih rendah menandakan model lebih akurat [15]. Dengan L_{CE} seperti pada Persamaan (5), exp (·) adalah eksponensial natural yang mengonversi loss ke skala per token.

$$PPL(X) = \exp\left(L_{CE}\right) \tag{6}$$

4. Sequence Exact Match

Proporsi keluaran identik dengan referensi, sebagai kriteria ketat untuk evaluasi generasi [16]. Notasi: N=jumlah sampel, $\hat{Y}_i$=urutan prediksi, Y_i=urutan target, dan [·] bernilai 1 jika kedua urutan sama, serta 0 jika berbeda.

$$EM_{sequence} = \frac{1}{N}\sum_{i=1}^{N}[\hat{Y}_i = Y_i] \tag{7}$$

5. BLEU Score (BLEU-4)

Menilai kesamaan antara urutan prediksi dan referensi berdasarkan kecocokan *n-gram* [17]. Notasi: BP=*brevity penalty*, w_n=bobot (1/4), p_n=presisi *n-gram*.

$$BLEU = BP \cdot exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{8}$$

6. ROUGE-L

Mengukur kesamaan urutan terpanjang (*Longest Common Subsequence*) antara hasil prediksi dan referensi [18]. Notasi: P=presisi LCS, R=recall LCS, β=bobot.

$$ROUGE - L = F_\beta = \frac{(1+\beta^2)\cdot P \cdot R}{\beta^2 \cdot P + R} \quad (9)$$

7. Normalized Edit Distance

Menghitung jumlah perubahan minimal antara prediksi dan referensi, dinormalisasi terhadap panjang urutan [19]. Notasi: Levenshtein=operasi *insert/delete/substitute*, | $\hat{Y}$ |, |Y|= panjang urutan.

$$ED_{norm} = \frac{Levenshtein(Y, \hat{Y})}{\max(|Y|, |\hat{Y}|)} \quad (10)$$

F. Analisis Hasil

Analisis hasil difokuskan pada perbandingan metrik kuantitatif antar varian model guna mengidentifikasi keunggulan dan *trade-off* arsitektur, disertai interpretasi kualitatif melalui inferensi pada sampel lirik dan progresi *chord* representatif untuk mengontekstualisasikan temuan secara keseluruhan.

IV. HASIL DAN PEMBAHASAN

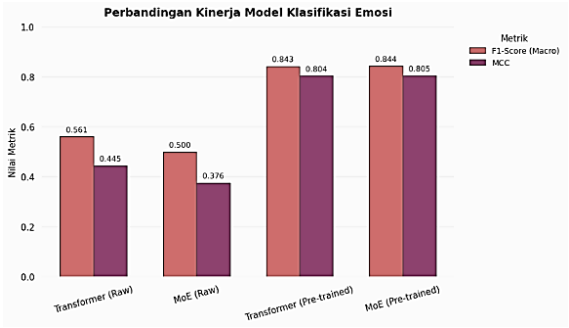
Implementasi sistem Hamisfera dan eksperimen perbandingan model dilakukan menggunakan *Python* dengan *framework PyTorch*, di mana model pralatih diakses melalui library Transformers *Hugging Face*. Pelatihan serta inferensi berjalan pada *Google Colaboratory* dengan GPU NVIDIA T4 untuk akselerasi komputasi, memastikan skalabilitas pada sumber daya terbatas sambil menjaga reprodusibilitas melalui *seed* acak tetap dan *logging* metrik per *epoch*.

A. Hasil Kinerja Model Klasifikasi Emosi (Model1)

Tahap pertama dalam Hamisfera, yaitu mengklasifikasikan emosi dari lirik lagu, pengujian dilakukan terhadap 4 varian model. Tujuannya untuk membandingkan peforma Transformer dengan MoE, serta melihat perbedaan saat menggunakan model pralatih (PT) dibandingkan memulai dari awal (RAW). Hasil setelah 100 epoch ada di Tabel 4, yang menunjukkan seberapa baik model menangani emosi multibahasa dengan label seperti *happy, sad, romantic, dark, dan hopeful*.

TABEL 4
HASIL KINERJA MODEL KLASIFIKASI

Metric	T-RAW	MoE-RAW	T-PT	MoE-PT
F1-score (Macro)	0.561	0.500	0.843	**0.844**
Accuracy	0.555	0.499	0.843	**0.844**
AUC (OVR)	0.817	0.769	**0.948**	0.935
MCC	0.445	0.376	0.804	**0.805**
Cohen's Kappa	0.444	0.374	0.804	**0.805**
Train Loss	0.610	0.852	**0.241**	0.437



Gambar 3 Perbandingan Kinerja Model Klasifikasi

Dari Tabel 4 dan Gambar 3, terlihat kesenjangan performa yang signifikan antara varian RAW dan PT. Model T-RAW dan MoE-RAW hanya mencapai *F1-Score macro* sekitar 0.50 sampai 0.56, sedangkan T-PT dan MoE-PT melonjak hingga 0.84. Perbedaan ini dapat dijelaskan melalui perspektif *transfer learning*, di mana model RAW harus membangun representasi semantik dari nol pada dataset terbatas sebanyak 73.369 entri yang tidak seimbang. Tanpa pengetahuan pralatih, model kesulitan menangkap pola linguistik halus, seperti sinonim emosional lintas bahasa ("sedih" dalam bahasa Indonesia, "*sad*" dalam bahasa Inggris, atau "*kanashii*" dalam bahasa Jepang), sehingga cenderung jatuh pada prediksi mayoritas dan mengabaikan nuansa budaya. Hal ini tercermin pada *train loss* yang tetap tinggi pada RAW (0.61–0.85) dibandingkan PT (0.24–0.44), menandakan konvergensi lambat dan potensi *underfitting* pada kelas minoritas. Sebaliknya, varian PT memanfaatkan representasi XLM-R yang telah dilatih pada korpus teks luas dari lebih dari 100 bahasa, sehingga mampu mentransfer pemahaman semantik dasar ke tugas emosi lirik, menghasilkan peningkatan *F1-Score*

pISSN: 2252-486X
eISSN: 2548-4710

hingga 60%. Menariknya, setelah *pre-training*, perbedaan antara Transformer dan MoE menjadi minimal (hanya 0.001 pada *F1-Score*), menunjukkan bahwa *bottleneck* utama adalah kualitas representasi awal, bukan arsitektur itu sendiri. Implikasinya, dalam konteks multibahasa, *pre-training* menjadi faktor krusial untuk menghasilkan performa yang kompetitif, terutama pada dataset berukuran sedang.

Secara kualitatif, *pre-training* XLM-R membantu interpretasi emosi lirik multibahasa melalui mekanisme representasi yang komprehensif. Secara fundamental, penyelarasan lintas bahasa yang konsisten membuat ungkapan emosi yang setara pada bahasa Indonesia, Inggris, dan Jepang berada pada wilayah representasi yang berdekatan, sehingga label emosi per baris dan per segmen menjadi lebih stabil meskipun terjadi variasi kosakata atau campuran bahasa. Selain itu, sensitivitas terhadap fenomena linguistik halus meningkat karena model telah terekspos beragam pola pada korpus besar, sehingga lebih peka terhadap sinonim, negasi, penguat intensitas, dan idiom budaya. Hal ini menyebabkan frasa yang secara permukaan serupa tetapi bernuansa emosional berbeda tidak lagi dipetakan ke kelas yang sama. Lebih jauh lagi, ketangguhan (*robustness*) terhadap variasi morfologi dan ejaan terbantu oleh tokenisasi *subword*, sehingga bentuk kata yang jarang atau terikat afiks tetap memunculkan sinyal semantik yang tepat tanpa masalah *out-of-vocabulary*.

Dampaknya, distribusi probabilitas pada keluaran klasifikasi menjadi lebih tajam, entropi menurun, dan penetapan label mayoritas per segmen lebih konsisten. Kondisi emosi yang lebih bersih ini kemudian memperbaiki tahap generatif karena Model 2 menerima kondisi yang jelas, sehingga progresi *chord* dalam notasi romawi lebih selaras dengan makna lirik dan transisi antarbirama cenderung lebih halus.

B. Hasil Kinerja Model Generasi Progresi Chord (Model 2)

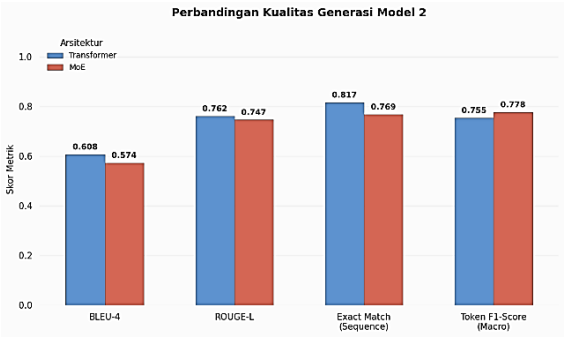
Untuk model generatif, pengujian berfokus pada kualitas sekuens dan efisiensi komputasi, sambil membandingkan Transformer dengan MoE dalam tugas membuat progresi *chord* secara otomatis.

1. Kualitas Generasi

Kualitas progresi *chord* dinilai dengan metrik standar di Tabel 5, yang mengukur kefasihan, kesesuaian bentuk, dan ketepatan meniru referensi.

TABEL 5
HASIL KINERJA MODEL GENERASI

Metric	Transformer	MoE
Perplexity	**2.124**	2.138
BLEU-4	**0.608**	0.574
ROUGE-L	**0.762**	0.747
Exact Match (Sequence)	**0.817**	0.769
Train Loss	**0.460**	0.500
Normalized Edit Distance	**0.273**	0.296



Gambar 4 Perbandingan Kualitas Generasi

Tabel 5 dan Gambar 4 menunjukkan Transformer lebih unggul dalam menghasilkan sekuens berkualitas, dengan nilai lebih baik di hampir semua ukuran. Nilai *perplexity* lebih rendah, serta *BLEU-4* dan *ROUGE-L* lebih tinggi yang menandakan kemampuannya menjaga alur harmonik yang panjang, seperti pola berulang, sehingga hasilnya terasa alami dan mirip referensi, terutama untuk emosi positif di mana *exact match* mencapai 82%. Pada tingkat *sequence*, MoE sedikit tertinggal yang kemungkinan dipicu oleh batasan *routing* dengan *top-k=2* per token sehingga cakupan konteks menyempit dan transisi antar *chord* kurang mulus. Di tingkat token untuk kelas emosi romantis, MoE lebih presisi, namun *training loss* yang cenderung lebih tinggi menandakan perlunya penyesuaian bobot *auxiliary loss* dan *hyperparameter gating* agar distribusi beban *expert* lebih seimbang. Dengan demikian, Transformer lebih tepat sebagai *backbone* generatif Hamisfera untuk menjaga kelancaran harmoni, sementara elemen MoE dapat difungsikan sebagai adapter opsional

untuk menambah variasi tanpa mengorbankan *fluency*, misalnya melalui peningkatan *top-k* atau *capacity factor* secara terkontrol.

2. Efisiensi Komputasi

Efisiensi diukur saat pelatihan (Tabel 6) dan inferensi (Tabel 7), melalui waktu dan penggunaan memori untuk melihat skalabilitasnya.

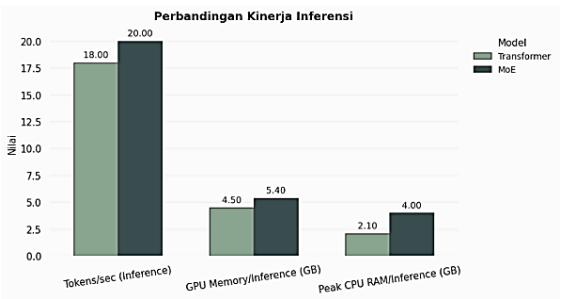
TABEL 6
BEBAN KOMPUTASI PELATIHAN

Metric	Transformer	MoE
Rata-rata waktu per *Epoch* (detik)	384	765
Rata-rata Memori GPU (GB)	7.7	11.6

Pada Tabel 6, Transformer lebih efisien selama pelatihan dengan satu *epoch* hampir dua kali lebih cepat dan pengguanaan memori GPU lebih rendah. Keunggulan ini muncul dari jalur komputasi yang padat dan deterministik tanpa *overhead routing top-k, capacity check, dan sparse dispatch* yang pada MoE menambah latensi sekitar 15% per iterasi, sehingga Transformer lebih tepat untuk uji coba cepat pada skala data ini, sedangkan MoE lebih bernilai pada skala data dan parameter yang lebih besar, saat aktivasi *sparse activation* benar-benar menurunkan komputasi efektif per *batch* sehingga kecepatan pemrosesan per langkah meningkat.

TABEL 7
BEBAN KOMPUTASI INFERENSI

Metric	Transformer	MoE
Tokens/sec (Inference)	18	**20**
GPU Memory/Inference (GB)	**4.5**	5.4
Peak CPU RAM/Inference (GB)	**2.1**	4.0



Gambar 5 Perbandingan Efisiensi Komputasi

Saat inferensi, Tabel 6 dan Gambar 5 menunjukkan MoE sedikit lebih cepat dalam *token generation speed*. Keunggulan ini muncul karena *sparse activation* mengurangi beban komputasi sekitar 20% pada mode evaluasi, tetapi biayanya cukup besar yakni penggunaan memori GPU naik sekitar 20% dan puncak RAM CPU hampir dua kali lipat akibat penyimpanan status per rute dan penanganan *overflow*. karena itu Transformer lebih ramah untuk perangkat dengan batas sumber daya, sementara MoE dapat diseimbangkan dengan menurunkan *top-k routing* atau mengoptimalkan *cache* agar *trade-off* kecepatan dan memori lebih baik.

C. Pengujian Kualitatif

Pengujian kualitatif ini bertujuan untuk memperkaya pemahaman tentang perilaku model generatif dalam menciptakan progresi *chord*, sebagai pelengkap metrik kuantitatif dengan penilaian musikal dan kontekstual. Melalui dua pendekatan inferensi menggunakan data uji dengan referensi *ground truth* dan lirik baru tanpa referensi, penilaian mencakup koherensi harmonik, relevansi emosional, dan kemampuan generalisasi, berdasarkan prinsip teori musik seperti siklus *chord* konvensional dan pergeseran tonal.

1. Pengujian dengan Data Uji (Ground Truth)

Pengujian ini membandingkan *output* model dengan progresi *chord* referensi pada sampel lirik dari data uji, untuk mengukur kedekatan struktural serta interpretasi musikal.

TABEL 8
UJI INFERENSI GROUND TRUTH

Model	lirik	Ground Truth	Hasil
Transformer	*the hardest parts the unspeakable thing that ive seen*	I - iii - iv - bVI	I - iii - V
MoE	*the hardest parts the unspeakable thing that ive seen*	I - iii - iv - bVI	I - iii - IV - vi

Semua simbol seperti I, iii, iv, bVI, V, IV, vi merupakan notasi romawi yang menunjukkan fungsi harmonik *chord* relatif terhadap kunci dasar lagu (bukan kunci absolut seperti C atau

pISSN: 2252-486X
eISSN: 2548-4710

Am), di mana huruf kapital (I, IV, V) menandakan *chord mayor*, huruf kecil (ii, iii, vi) menandakan *chord* minor, serta awalan *b* atau # menunjukkan *accidental* (penurunan atau penaikan setengah nada), misalnya, I adalah *tonic mayor*, iii adalah *mediant minor*, iv adalah *subdominant minor*, bVI adalah *submediant* yang diturunkan setengah nada untuk efek dramatis, V adalah *dominant* yang mendorong resolusi, IV adalah *subdominant mayor* untuk transisi hangat, dan vi adalah *relative minor* untuk variasi emosional.

Pada Tabel 8, progresi referensi (I - iii - iv - bVI) mencerminkan pola harmonik kompleks yang sering ditemui dalam musik dramatis, di mana perpindahan ke *chord minor* dan penggunaan bVI membangun ketegangan emosional yang mendalam, cocok dengan lirik yang penuh beban psikologis. Transformer menghasilkan *output* I - iii - V, pola sederhana yang langsung beresolusi ke *dominant mayor*, sehingga kurang mencerminkan intensitas lirik. Sebaliknya, MoE menghasilkan *output* I - iii - IV - vi, yang mempertahankan panjang empat *chord* dan nuansa melankolis melalui IV dan vi, sehingga lebih sesuai dengan tema "*unspeakable"* (yang sulit diungkapkan). Hasil ini menunjukkan bahwa MoE lebih unggul dalam menangkap struktur harmonik yang utuh dan kontekstual, meski masih kesulitan meniru modulasi seperti bVI.

2. Pengujian dengan Data Baru (Tanpa Ground Truth)

Pengujian ini menyelidiki kreativitas model pada lirik orisinal di luar dataset pelatihan, dengan penekanan pada interpretasi emosional dan inovasi harmonik secara subjektif.

TABEL 9
UJI INFERENSI DATA BARU

Model	Lirik	Hasil
Transformer	*Nymphaea blooms where silence falls,* *A fragile light above it all.* *She holds her calm beneath the rain,* *And lets the night erase the pain.*	IV - V - I - iii
MoE	*Nymphaea blooms where silence falls,* *A fragile light above it all.* *She holds her calm beneath the rain,* *And lets the night erase the pain.*	i - VII - VI

Lirik pada Tabel 9 memiliki ambiguitas nuansa (*dual-valence*), memadukan elemen harapan ("*blooms*", "*light*") dengan atmosfer melankolis ("*rain*", "*night*"). Transformer menghasilkan output IV - V - I - iii dalam modus Mayor yang menawarkan interpretasi optimis namun kontemplatif, pergerakan kadens IV-V-I memberikan resolusi terang untuk menggambarkan sisi harapan, namun secara cerdas diakhiri *chord* iii (*mediant minor*) yang lembut untuk menjaga kesan tenang. Sebaliknya, MoE menghasilkan pola menurun i - VII - VI dalam modus minor *Aeolian* yang gelap dan menggantung, lebih fokus menangkap sisi suram dari kata "*pain*" dan "*night*". Perbedaan ini mengindikasikan bahwa Transformer cenderung menjaga stabilitas resolusi tonal (konsisten dengan struktur lagu populer), sementara MoE lebih sensitif terhadap deteksi kata kunci emosi negatif dalam lirik.

3. Pengujian dengan decoding yang berbeda

Untuk memvalidasi keputusan penggunaan *greedy decoding* pada sistem Hamisfera, penelitian ini melakukan pengujian komparatif terhadap strategi inferensi *sampling* dan *beam search*. Pengujian dilakukan pada segmen lirik dengan label emosi prediksi "*Hopeful*". Tujuannya adalah melihat apakah variasi strategi decoding dapat meningkatkan kualitas progresi tanpa merusak relevansi emosi. Hasil perbandingan disajikan pada Tabel 10.

TABEL 10
PERBANDINGAN HASIL PROGRESI BERDASARKAN STRATEGI DECODING

Strategi	Parameter	Hasil
Greedy	-	IV - V - I - iii
Beam	*k=5*	IV - V - I - iii
Sampling(1)	*p=0.95, temp=1.1*	bVI - V - #iii
Sampling(2)	*p=0.95, temp=1.1*	V7 - I - iii - II

Berdasarkan Tabel 10, analisis kualitatif menunjukkan perbedaan fundamental dalam kualitas musikalitas antarstrategi. *Greedy Search* menghasilkan progresi IV - V - I - iii, sebuah pola yang diawali dengan resolusi kadens standar (IV-V-I) dan diakhiri oleh *mediant chord* (iii). Struktur ini sangat koheren dan diatonis (sesuai kunci), di mana *chord* iii

berfungsi memperpanjang alur harmonik dengan nuansa lembut yang relevan dengan emosi "*Hopeful*". Sementara itu, hasil *Beam Search* yang identik dengan *Greedy* mengindikasikan bahwa distribusi probabilitas model sudah sangat tajam pada sekuens optimal tersebut, sehingga pencarian jalur alternatif tidak memberikan variasi signifikan selain penambahan beban komputasi.

Sebaliknya, penerapan strategi stokastik melalui *Nucleus Sampling* justru memunculkan fenomena halusinasi harmonik dan inkonsistensi tonal. Pada *Sampling (1)*, muncul *chord* non-standar #iii dan *chord* bVI (*submediant mol*) yang menciptakan disonansi tajam dan nuansa gelap, bertentangan dengan konteks lirik. Sementara pada *Sampling (2)*, muncul *chord* II (*Mayor Supertonic*) yang bersifat kromatis, meskipun valid secara teori sebagai *secondary dominant*, penempatannya di akhir sekuens tanpa resolusi menciptakan ketegangan yang tidak perlu. Ketidakstabilan ini menegaskan bahwa hilangnya kontrol deterministik berisiko menghasilkan *output* yang atonal atau melanggar sintaksis harmoni sederhana. Oleh karena itu, pendekatan deterministik dikonfirmasi sebagai pilihan paling aman dan valid untuk sistem Hamisfera.

D. Analisis dan Rekomendasi Konfigurasi Hibrida

Berdasarkan evaluasi kinerja, penelitian ini merekomendasikan konfigurasi hibrida yang diimplementasikan melalui mekanisme *pipeline* sekuensial dua tahap. Pada tahap pertama, MoE-PT difungsikan sebagai *semantic filter* untuk mengonversi lirik mentah menjadi label emosi yang tegas, memanfaatkan keunggulan akurasinya (*F1-Score* 0.844) dalam menangani variasi linguistik. Prediksi label emosi ini kemudian dimasukkan sebagai token kondisi (*conditioning token*) eksplisit ke dalam Transformer-PT yang bertindak sebagai acuan semantik dalam pembentukan sekuens musik.

Pemilihan Transformer-PT sebagai generator akhir didasarkan pada kemampuannya dalam menjaga stabilitas struktur musik (BLEU-4 0.608) dan efisiensi komputasi dibandingkan MoE generatif. Sinergi ini terbukti paling optimal karena membagi beban tugas secara spesifik, MoE menangani ambiguitas semantik lirik yang kompleks, sementara Transformer menjamin koherensi harmonik progresi *chord*. Pendekatan ini secara efektif meminimalkan propagasi kesalahan (*error propagation*) yang sering terjadi pada model tunggal *end-to-end*, menghasilkan sistem yang seimbang antara ketepatan interpretasi emosi dan kualitas musikalitas.

V. KESIMPULAN

Penelitian ini telah menjawab pertanyaan utama mengenai arsitektur *deep learning* yang paling sesuai untuk sistem Hamisfera melalui perbandingan komprehensif. Hasil eksperimen menunjukkan bahwa konfigurasi hibrida yang mengintegrasikan prediksi emosi MoE sebagai input kondisi bagi Transformer memberikan keseimbangan terbaik.

Secara spesifik, untuk tahap klasifikasi emosi lirik (Model 1), arsitektur MoE pralatih direkomendasikan karena keunggulannya yang konsisten pada metrik evaluasi utama, seperti *F1-Score macro* (0,844), yang mencerminkan kestabilan dalam menangani variasi multibahasa tanpa mengorbankan akurasi. Sementara itu, untuk tahap generasi progresi *chord* (Model 2), Transformer pralatih terbukti lebih unggul, dengan skor BLEU-4 (0,608) dan ROUGE-L (0,762) yang lebih tinggi, serta efisiensi sumber daya yang mendukung penerapan praktis di lingkungan terbatas. Sinergi ini memastikan interpretasi lirik yang akurat diterjemahkan menjadi struktur musik yang koheren.

Namun, penelitian ini memiliki beberapa keterbatasan yang layak dicatat untuk pengembangan lebih lanjut. Dataset yang digunakan, meskipun telah diaugmentasi, masih berpotensi mengandung bias subjektif dalam pelabelan emosi yang dapat memengaruhi generalisasi. Evaluasi kualitas generasi juga masih bergantung pada metrik berbasis referensi, yang tidak selalu sepenuhnya mewakili penilaian manusia terhadap estetika musikalitas. Oleh karena itu, penelitian mendatang disarankan untuk melibatkan studi pengguna skala besar dengan partisipan musisi guna mendapatkan umpan balik subjektif, serta memperluas dataset ke genre yang lebih beragam untuk meningkatkan adaptabilitas sistem.

pISSN: 2252-486X
eISSN: 2548-4710

REFERENSI

[1] D.-V.-T. Le, L. Bigo, D. Herremans, and M. Keller, "Natural Language Processing Methods for Symbolic Music Generation and Information Retrieval: A Survey," *ACM Comput. Surv.*, vol. 57, no. 7, pp. 1–40, July 2025, doi: 10.1145/3714457.

[2] C. Zhang *et al.*, "ReLyMe: Improving Lyric-to-Melody Generation by Incorporating Lyric-Melody Relationships," in *Proceedings of the 30th ACM International Conference on Multimedia*, Oct. 2022, pp. 1047–1056. doi: 10.1145/3503161.3548357.

[3] Z. Ju *et al.*, "TeleMelody: Lyric-to-Melody Generation with a Template-Based Two-Stage Method," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds., Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 5426–5437. doi: 10.18653/v1/2022.emnlp-main.364.

[4] A. Vaswani *et al.*, "Attention is All you Need," in *Neural Information Processing Systems*, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:13756489

[5] K. Choi, J. Park, W. Heo, S. Jeon, and J. Park, "Chord Conditioned Melody Generation With Transformer Based Decoders," *IEEE Access*, vol. 9, pp. 42071–42080, 2021, doi: 10.1109/ACCESS.2021.3065831.

[6] Y.-H. Lan, W.-Y. Hsiao, H.-C. Cheng, and Y.-H. Yang, "MusiConGen: Rhythm and Chord Control for Transformer-Based Text-to-Music Generation," in *International Society for Music Information Retrieval Conference*, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:271328459

[7] B. Yu *et al.*, "Museformer: Transformer with Fine- and Coarse-Grained Attention for Music Generation," Oct. 31, 2022, *arXiv*: arXiv:2210.10349. doi: 10.48550/arXiv.2210.10349.

[8] A. Conneau *et al.*, "Unsupervised Cross-lingual Representation Learning at Scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online: Association for Computational Linguistics, 2020, pp. 8440–8451. doi: 10.18653/v1/2020.acl-main.747.

[9] A. Q. Jiang *et al.*, "Mixtral of Experts," Jan. 08, 2024, *arXiv*: arXiv:2401.04088. doi: 10.48550/arXiv.2401.04088.

[10] N. Shazeer *et al.*, "Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer," Jan. 23, 2017, *arXiv*: arXiv:1701.06538. doi: 10.48550/arXiv.1701.06538.

[11] Y. Hong and Y. Xue, "Emotion Classification through Song Lyrics in Multi-Languages with Bert," *Applied and Computational Engineering*, 2025, [Online]. Available: https://api.semanticscholar.org/CorpusID:276694176

[12] Z. Sheng *et al.*, "SongMASS: Automatic Song Writing with Pre-training and Alignment Constraint," in *AAAI Conference on Artificial Intelligence*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:228064027

[13] P. Vickers, L. Barrault, E. Monti, and N. Aletras, "We Need to Talk About Classification Evaluation Metrics in NLP," in *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, Nusa Dua, Bali: Association for Computational Linguistics, 2023, pp. 498–510. doi: 10.18653/v1/2023.ijcnlp-main.33.

[14] A. Mao, M. Mohri, and Y. Zhong, "Cross-Entropy Loss Functions: Theoretical Analysis and Applications," June 20, 2023, *arXiv*: arXiv:2304.07288. doi: 10.48550/arXiv.2304.07288.

[15] G. Alon and M. Kamfonas, "Detecting Language Model Attacks with Perplexity," Nov. 07, 2023, *arXiv*: arXiv:2308.14132. doi: 10.48550/arXiv.2308.14132.

[16] M. İncidelen and M. Aydoğan, "Performance Evaluation of Transformer-Based Pre-Trained Language Models for Turkish Question-Answering," *Black Sea Journal of Engineering and Science*, vol. 8, no. 2, pp. 323–329, Mar. 2025, doi: 10.34248/bsengineering.1596832.

[17] P. Rust, B. Shi, S. Wang, N. C. Camgöz, and J. Maillard, "Towards Privacy-Aware Sign Language Translation at Scale," Aug. 07, 2024, *arXiv*: arXiv:2402.09611. doi: 10.48550/arXiv.2402.09611.

[18] Y. Yuniati, K. M. Fitria, Melvi, S. Purwiyanti, E. Nasrullah, and M. A Muhammad, "Analisis Performa Ekstraksi Konten GPT-3 Dengan Matrik Bertscore Dan Rouge," *JTIIK*, vol. 11, no. 6, pp. 1273–1280, Dec. 2024, doi: 10.25126/jtiik.1168088.

[19] D. Fisman and I. Tzarfati, “When Is the Normalized Edit Distance over Non-Uniform Weights a Metric?,” *LIPIcs, Volume 296, CPM 2024*, vol. 296, p. 14:1-14:17, 2024, doi: 10.4230/LIPICS.CPM.2024.14.

pISSN: 2252-486X
eISSN: 2548-4710