Pengembangan Sistem Text-to-Speech Berbahasa Indonesia dengan Kontrol Karakteristik berbasis Prompt

M.Alif Al Hakim[1], Winoto Hasyim[2], Matthew Hotmaraja Johan Turnip[3]
[1]Fakultas Ilmu Komputer Universitas Indonesia, Depok, 16424, email: malif.al@ui.ac.id
[2]Fakultas Ilmu Komputer Universitas Indonesia, Depok, 16424, email: winoto.hasyim@ui.ac.id
[3]Fakultas Ilmu Komputer Universitas Indonesia, Depok, 16424, email: matthew.hotmaraja@ui.ac.id

INTISARI — Sistem *Text-to-Speech* (TTS) berbasis *prompt* menawarkan kemampuan menghasilkan audio dengan karakteristik suara yang dapat disesuaikan sesuai deskripsi teks. Teknologi ini berpotensi mendukung berbagai aplikasi, seperti pembelajaran bahasa dan aksesibilitas informasi. Namun, pengembangan model TTS untuk bahasa Indonesia menghadapi tantangan, terutama dalam menghasilkan audio berkualitas tinggi yang sesuai dengan deskripsi teks. Masalah ini diperburuk oleh keterbatasan dataset berbahasa Indonesia dengan variasi suara yang memadai. Penelitian ini mengusulkan Parler-TTS, model berbasis *prompt* yang terdiri dari encoder teks untuk menangkap informasi semantik, decoder autoregresif dengan mekanisme *cross-attention* untuk menghasilkan token audio, dan DAC sebagai *codec audio* untuk rekonstruksi gelombang suara. Model dilatih pada *dataset* audio bahasa Indonesia yang dikumpulkan melalui Youtube. Sebelum digunakan untuk melatih model, *dataset* dilakukan preprocessing dan anotasi secara otomatis terlebih dahulu menggunakan beberapa pendekatan untuk mengukur kualitas dan karakteristik audio. Model dilatih menggunakan inisialisasi parameter dari model berbahasa Inggris untuk mempercepat konvergensi. Penyesuaian hyperparameter juga dilakukan untuk meningkatkan performa. Hasil eksperimen menunjukkan bahwa skor *CLAP similarity* (0,21) dan *Word Error Rate* (51%) masih tergolong rendah. Namun, secara subjektif model menghasilkan audio dengan skor MOS yang cukup baik untuk kejelasan (3,825) dan relevansi terhadap prompt (4,375). Penelitian ini memberikan kontribusi dalam bentuk pengembangan model TTS berbasis prompt yang dapat disesuaikan, serta dataset audio bahasa Indonesia yang komprehensif, membuka peluang untuk pengembangan lebih lanjut.

KATA KUNCI — *Text-to-Speech*, Prompt, Karakteristik, Suara, *Cross-Attention*

I. PENDAHULUAN

Teknologi Text-to-Speech (TTS) telah mengalami perkembangan pesat dalam beberapa tahun terakhir dengan berbagai aplikasi yang mencakup pendidikan, hiburan, dan *assistive technology*. Dalam penelitian ini, kami mengeksplorasi pengembangan sistem TTS berbasis kontrol karakteristik suara menggunakan deskripsi teks atau *prompt* untuk menghasilkan audio yang sesuai dengan kebutuhan pengguna. Dengan pendekatan ini, pengguna dapat menentukan atribut seperti kecepatan bicara, intonasi, dan nada secara langsung melalui teks deskriptif.

Meskipun berbagai model TTS telah mencapai tingkat kealamian dan kejelasan yang baik, kemampuan untuk mengontrol karakteristik suara secara spesifik melalui *prompt* teks masih terbatas. Masalah utama terletak pada terbatasnya dataset berbahasa Indonesia dengan variasi suara yang memadai, yang sering kali mengakibatkan kurang optimalnya kinerja model dalam menghasilkan audio dengan atribut yang akurat sesuai deskripsi.

Penelitian ini mengusulkan model Parler-TTS, sistem TTS berbasis *prompt* yang dirancang untuk menghasilkan audio sesuai deskripsi pengguna. Model ini terdiri atas tiga komponen utama: *text encoder*, *decoder*, dan audio codec. *Text encoder* menggunakan arsitektur Flan-T5 untuk mengubah deskripsi teks menjadi representasi semantik. Representasi tersebut digunakan untuk menginisiasi generasi token audio pada dekoder autoregresif yang dilengkapi dengan mekanisme *cross-attention*. Mekanisme ini memastikan informasi semantik dari teks dimanfaatkan secara optimal selama proses generasi suara, sehingga karakteristik audio yang dihasilkan konsisten dengan deskripsi pengguna. Komponen terakhir adalah audio codec berbasis *Descript Audio Codec* (DAC), yang berfungsi merekonstruksi gelombang suara dari token audio.

Proses pelatihan dilakukan dengan menggunakan dataset audio berbahasa Indonesia yang telah dianotasi dan diproses secara ekstensif. Langkah-langkah pemrosesan meliputi normalisasi teks, ekstraksi fitur suara, serta pembuatan deskripsi berbasis kategori karakteristik suara. Untuk mempercepat konvergensi, model diinisialisasi dengan parameter yang telah dilatih menggunakan dataset berbahasa Inggris sebelum dilanjutkan dengan *fine-tuning* pada data bahasa Indonesia. *Hyperparameter tuning* juga diterapkan untuk mengoptimalkan stabilitas dan efisiensi pelatihan, dengan penyesuaian pada *learning rate*, *batch size*, dan jumlah *epoch*.

Evaluasi dilakukan menggunakan metrik seperti Mean Opinion Score (MOS) untuk kualitas audio, CLAP similarity untuk kesesuaian audio dengan deskripsi, dan Word Error Rate (WER) untuk akurasi pengucapan kata. Hasil eksperimen menunjukkan bahwa model mampu menghasilkan audio dengan nilai MOS sebesar 3.825 untuk kejelasan dan 4.375 untuk kesesuaian dengan deskripsi teks. Namun, performa pada metrik objektif seperti CLAP similarity (0.21) dan WER (51%) masih memerlukan perbaikan. Hasil ini menunjukkan potensi model untuk ditingkatkan lebih lanjut dengan dataset yang lebih beragam, berkualitas tinggi, dan pelatihan yang lebih lama.

Penelitian ini memberikan kontribusi utama dalam pengembangan sistem TTS berbasis *prompt* yang memungkinkan kontrol karakteristik suara pada bahasa Indonesia. Metodologi yang diajukan mencakup integrasi arsitektur model yang inovatif dengan pemrosesan dataset yang komprehensif, serta evaluasi performa menggunakan metrik subjektif dan objektif. Penelitian ini membuka peluang baru untuk pengembangan sistem TTS yang lebih adaptif dan sesuai dengan kebutuhan lokal.

Laporan ini diorganisasikan sebagai berikut: Bagian I membahas pengantar dan tujuan penelitian. Bagian II merangkum kajian pustaka terkait. Bagian III menjelaskan metodologi yang diusulkan, termasuk arsitektur model dan proses pelatihan. Bagian IV memaparkan eksperimen dan analisis hasil. Bagian V menyimpulkan temuan utama dan memberikan rekomendasi untuk penelitian di masa depan.

II. Kajian Pustaka

Pengembangan sistem TTS dengan kemampuan kontrol berbasis *prompt* telah menjadi bidang penelitian yang semakin penting untuk memenuhi kebutuhan fleksibilitas dan kualitas audio yang tinggi. Penelitian oleh Lyth dan King [1] memperkenalkan pendekatan berbasis *prompt* untuk mengontrol karakteristik suara, seperti identitas pembicara, gaya berbicara, dan kondisi akustik. Model mereka memanfaatkan anotasi sintetik pada dataset berskala besar (45 ribu jam) untuk melatih model bahasa ucapan. Pendekatan ini memungkinkan pengendalian intuitif melalui deskripsi dalam bahasa natural, menghasilkan ucapan berkualitas tinggi tanpa memerlukan pelabelan manual dalam skala besar.

Berbeda dengan metode tradisional seperti *global style tokens* [2] atau *reference embeddings* [3] yang bergantung pada rekaman referensi untuk mengontrol gaya suara, pendekatan berbasis *prompt* yang digunakan oleh Lyth dan King memberikan fleksibilitas yang lebih besar. Dengan menggantikan anotasi manual dengan anotasi otomatis, mereka mampu menggunakan dataset dalam skala besar tanpa menghadapi hambatan pelabelan manusia. Sistem berbasis prompt lainnya, seperti Audiobox [4] dan PromptTTS2 [5], juga telah mengeksplorasi penggunaan deskripsi natural, tetapi kemampuan kontrolnya terbatas pada atribut tertentu seperti emosi atau kecepatan bicara. Dataset yang lebih kecil pada sistem tersebut juga membatasi skala dan fleksibilitas dibandingkan dengan model yang diusulkan oleh Lyth dan King.

Selain itu, penelitian lain menunjukkan potensi besar dalam generasi berbasis *prompt*, seperti yang ditunjukkan oleh MusicGen [6]. MusicGen menggunakan kombinasi deskripsi tekstual dan fitur melodi untuk menghasilkan musik yang dapat disesuaikan, dengan inovasi seperti MusicGen-Chord dan MusicGen-Remixer. Pendekatan modular ini menunjukkan bagaimana kontrol berbasis prompt dapat diterapkan untuk tugas generatif dengan tingkat presisi tinggi, sebuah gagasan yang relevan untuk TTS.

Namun, sebagian besar penelitian TTS berbasis *prompt* masih terbatas pada Bahasa Inggris. Misalnya, LibriTTS-P [7] menyediakan dataset beranotasi untuk Bahasa Inggris dengan deskripsi gaya berbicara dan karakteristik pembicara. Dataset ini memungkinkan model untuk menghasilkan ucapan yang lebih natural dan relevan dengan deskripsi teks. Hingga saat ini, pendekatan serupa belum diperluas ke Bahasa Indonesia, yang menghadirkan tantangan unik seperti struktur morfologi yang kompleks, fonologi khas, dan pola intonasi yang berbeda.

Penelitian ini bertujuan untuk mengadaptasi model berbasis *prompt* ke dalam Bahasa Indonesia dengan mengatasi tantangan linguistik yang ada. Sistem yang dikembangkan memanfaatkan arsitektur *encoder-decoder* dengan mekanisme *cross-attention* untuk menghasilkan ucapan yang sesuai dengan deskripsi teks. Inovasi utama dari penelitian ini mencakup integrasi anotasi sintetik untuk menciptakan dataset lokal yang mencerminkan kebutuhan budaya dan linguistik Bahasa Indonesia. Dengan mengadaptasi pendekatan modular seperti yang digunakan dalam MusicGen dan metode berbasis *prompt* dari Lyth dan King, penelitian ini berupaya memberikan solusi yang lebih relevan untuk kebutuhan lokal dan memperluas cakupan aplikasi TTS berbasis *prompt*.

III. Metodologi

A. *Model Parler TTS*

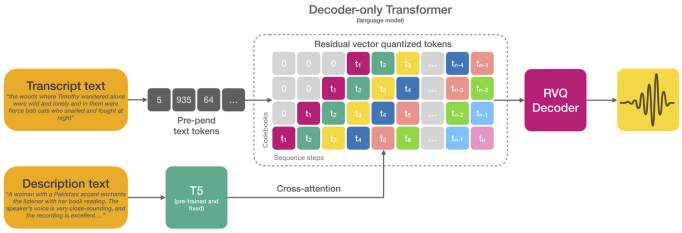
Model Parler-TTS [1] memiliki arsitektur yang mirip dengan arsitektur model MusicGen [8]. Model ini memiliki tiga bagian utama, diantaranya sebagai berikut

1) ***Text encoder***: Bagian ini berfungsi untuk mengubah deskripsi prompt mengenai karakteristik suara menjadi representasi hidden-state. Parler-TTS menggunakan sebuah frozen text encoder yang diinisialisasi dari model Flan-T5 [9].
2) ***Parler-TTS decoder***: Bagian ini adalah *language model* (LM) yang secara autoregresif menghasilkan token audio (atau audio codes) berdasarkan kondisi representasi *hidden state* yang dihasilkan oleh *text encoder*. Dekoder ini dirancang untuk menggunakan mekanisme *cross-attention* agar dapat menggunakan informasi dari representasi teks selama proses generasi audio.
3) ***Audio codec***: Bagian ini digunakan untuk memulihkan bentuk gelombang audio dari token audio yang diprediksi oleh dekoder. Parler-TTS menggunakan model DAC (*Descript Audio Codec*) [10] untuk bagian ini.

Meskipun arsitektur Parler-TTS mirip dengan MusicGen, ada beberapa modifikasi yang diperkenalkan, diantaranya penggunaan *cross-attention* untuk memperkaya keluaran audio dengan informasi deskripsi teks. Deskripsi teks yang telah diproses akan digabungkan dengan *hidden states* masukan dekoder atau kalimat yang ingin dijadikan audio. Selain itu, Parler-TTS menggunakan DAC sebagai audio *encoder* karena memberikan kualitas suara yang lebih baik dibandingkan dengan EnCodec [11].

B. *Cara kerja model Parler TTS*

Model Parler-TTS dirancang untuk menghasilkan audio dengan karakteristik suara yang dapat dikontrol menggunakan deskripsi teks berbasis prompt. Cara kerja model ini dimulai



Gambar 1: Gambaran umum arsitektur model Parler-TTS

dengan memproses input berupa deskripsi teks yang mendetail, seperti "*In a calm and slow voice, with a slight echo in the background*". Deskripsi ini diberikan ke komponen *Text Encoder*, yang bertugas mengubah teks menjadi representasi *hidden-state*. Parler-TTS menggunakan Flan-T5, sebuah encoder berbasis Transformer yang telah dilatih sebelumnya, sehingga mampu menangkap informasi semantik dari deskripsi teks secara efisien. Representasi *hidden-state* yang dihasilkan kemudian menjadi kondisi awal untuk proses generasi token audio.

Pada tahap selanjutnya, representasi hidden-state ini dimasukkan ke dalam Decoder, sebuah language model autoregresif yang secara bertahap menghasilkan token audio. Decoder Parler-TTS menggunakan mekanisme cross-attention, yang memungkinkan model untuk mengintegrasikan informasi dari deskripsi teks selama proses generasi audio. Dengan mekanisme ini, model mampu memastikan bahwa karakteristik suara yang dihasilkan, seperti kecepatan bicara, intonasi, reverberasi, dan nada, sesuai dengan deskripsi yang diberikan pengguna. Proses ini dilakukan secara autoregresif, di mana setiap token audio diprediksi berdasarkan token-token sebelumnya hingga seluruh rangkaian token audio selesai.

Tahap terakhir adalah rekonstruksi gelombang suara dari token audio yang dihasilkan. Parler-TTS menggunakan DAC, sebuah model *audio codec* yang dirancang untuk menghasilkan gelombang suara berkualitas tinggi dengan *sampling rate* hingga 44 kHz. DAC mengambil token audio dari decoder dan mengubahnya menjadi file audio akhir yang dapat didengar. Dengan arsitektur dan alur kerja ini, Parler-TTS mampu menghasilkan audio yang tidak hanya realistis tetapi juga sesuai dengan karakteristik yang diinginkan pengguna.

Pada proses *training*, akan digunakan fitur yang berasal dari *prompt* dan representasi *hidden-state* serta token audio yang dihasilkan dari audio *groundtruth* yang dimiliki. Setelah itu, akan dilakukan proses pelatihan pada model *decoder* yang juga menggunakan *cross-attention* untuk menemukan hubungan token audio dan representasi *hidden-state*. Model menghitung *attention score* untuk setiap pasangan token audio dan representasi *hidden-state*. *Attention score* ini menunjukkan seberapa relevan setiap token audio dengan setiap bagian dari representasi *hidden-state*. *Attention score* digunakan untuk memboboti representasi *hidden-state* terkait dengan suatu token audio. Bobot yang lebih tinggi diberikan pada bagian-bagian yang lebih relevan. Pelatihan menggunakan fungsi loss *cross-entropy* yang bertujuan untuk meminimalkan kesalahan antara prediksi token audio dan token audio target.

IV. Eksperimen

A. *Dataset Preparation and Processing*

Dataset yang digunakan untuk melatih dan menguji model adalah kumpulan audio berbahasa Indonesia yang diperoleh dari video YouTube milik Eno Bening dan Justinus Lhaksana.

1) **Pemerolehan Transkrip**
   Transkrip diperoleh secara otomatis menggunakan *script* yang memanfaatkan layanan *speech-to-text* berbasis API. Output dari proses ini berupa dua jenis file, yaitu file audio penuh yang berisi seluruh rekaman suara, dan kedua, sebuah file SRT yang berisi *timestamp* dan transkrip teks yang sesuai dengan bagian-bagian tertentu dari audio. File SRT ini mengidentifikasi rentang waktu tertentu dalam rekaman audio, dengan setiap segmen audio yang telah dianotasi dengan transkrip yang sesuai untuk rentang timestamp tersebut.
2) **Organisasi Data Audio**
   Masing-masing bagian audio yang terpisah berdasarkan *timestamp* pada file SRT akan diekstrak menjadi file audio terpisah dengan ID atau nama file unik tersendiri. Setelah itu, beberapa bagian atau *clip* audio akan digabungkan sehingga berdurasi sekitar 8 - 12 detik. Nama file digunakan untuk mengidentifikasi dan menyimpan setiap segmen audio terpisah. Proses ini bertujuan untuk mempermudah pengelolaan dan pemrosesan file audio yang lebih kecil. Selanjutnya, untuk mengorganisasi data audio dan transkripnya, sebuah file CSV akan dibuat. Setiap baris pada file CSV akan berisi nama file audio dan transkrip yang sesuai dengan segmen audio file tersebut. Proses ini memungkinkan untuk mengelompokkan dan memproses segmen audio yang lebih kecil dan terfokus, sehingga model dapat lebih mudah mempelajari karakteristik suara dalam konteks waktu tertentu. Total *clip* audio yang berhasil dikumpulkan adalah 25.550 atau setara dengan 65 jam.
3) **Normalisasi Teks Transkrip**
   Normalisasi teks bertujuan untuk mengubah teks mentah menjadi format yang lebih seragam dan siap untuk diproses oleh model TTS. Proses ini meliputi beberapa langkah, diantaranya ekspansi angka, *lower casting*, penghapusan tanda baca tidak berguna atau berlebih, dan penghapusan spasi berlebih. Hal ini diperlukan untuk memastikan bahwa input teks yang diberikan ke model TTS berada dalam bentuk yang optimal dan dapat diproses secara konsisten, sehingga meningkatkan akurasi dan kualitas suara yang dihasilkan.
4) **Ekstrak Fitur Karakteristik Suara**
   Untuk menghasilkan karakteristik suara yang bisa dikustomisasi melalui prompt, setiap audio perlu diekstrak karakteristik suaranya terlebih dahulu. Karakteristik suara yang diekstrak, diantaranya *speaking rate, recording quality, speech quality,* dan *utterance pitch*.
   a) ***Speaking Rate***
      Merujuk pada kecepatan berbicara dalam audio,

yang mempengaruhi seberapa cepat atau lambat ujaran disampaikan. Pengekstrakan *speaking rate* penting untuk memungkinkan penyesuaian tempo dalam output suara. Dihitung dengan membagi banyak *phoneme* dengan panjang *utterance*/ujaran.Pada penelitian ini, digunakan *library* Grapheme-to-Phoneme berbahasa indonesia (G2P-id) [12].

b) ***Recording Quality***
Kualitas rekaman audio dapat diukur secara objektif menggunakan parameter SNR (*Signal-to-Noise Ratio*) dan C50. SNR menunjukkan perbandingan antara kekuatan sinyal suara dengan noise latar belakang, sedangkan C50 mengukur waktu reverberasi atau gema dalam suatu ruangan. Model deteksi aktivitas suara seperti Brouhaha dari pyannote [13] dapat digunakan untuk menghitung kedua parameter ini. Nilai SNR yang tinggi mengindikasikan sedikitnya noise, sementara nilai C50 yang rendah menunjukkan sedikitnya gema. Informasi ini sangat berguna untuk mengevaluasi kualitas rekaman.

c) ***Speech Quality***
Untuk mengevaluasi kualitas suatu sinyal ucapan secara objektif, dapat dimanfaatkan berbagai metrik yang disediakan oleh Torchaudio. Torchaudio Speech Quality and Intelligibility Measures [14] menawarkan serangkaian alat untuk menghitung metrik-metrik penting seperti PESQ, STOI, dan SD-SDR.

i) **Perceptual Estimation of Speech Quality (PESQ):** Metrik PESQ dirancang untuk meniru persepsi manusia terhadap kualitas ucapan. PESQ memberikan skor numerik antara -0.5 hingga 4.5, di mana skor yang lebih tinggi menunjukkan kualitas yang lebih baik. PESQ sangat sensitif terhadap distorsi yang memengaruhi inteligibilitas dan naturalitas ucapan

ii) **Short-Time Objective Intelligibility (STOI):** STOI berfokus pada aspek inteligibilitas ucapan, yaitu seberapa mudah suatu ucapan dapat dipahami. STOI menghitung korelasi antara spektogram ucapan yang terdegradasi dengan spektogram ucapan bersih. Nilai STOI berkisar antara 0 hingga 1, di mana nilai yang mendekati 1 menunjukkan inteligibilitas yang tinggi.

iii) **Scale-Invariant Signal-to-Distortion Ratio (SD-SDR):** SD-SDR adalah metrik yang lebih baru dan dianggap lebih robust dibandingkan metrik tradisional seperti SDR. SD-SDR mengukur seberapa banyak sinyal asli yang masih tersisa dalam sinyal yang terdegradasi, sambil memperhitungkan skalanya. Nilai SD-SDR yang lebih tinggi menunjukkan lebih sedikit distorsi.

d) ***Pitch Utterance***
Pada penelitian ini, dianalisis *pitch utterance* atau nada rata-rata dari setiap *clip* audio. *Pitch* merupakan karakteristik akustik yang menentukan tinggi rendahnya suatu suara. Dengan mengukur *pitch utterance*, dapat diperoleh informasi mengenai intonasi pembicara. Untuk menghitung *pitch utterance* pada setiap *clip* audio, digunakan model *Pitch-Estimating Neural Networks* (PENN) [15]. Setelah memperoleh nilai *pitch* untuk setiap titik waktu dalam *clip* audio, akan dihitung dua statistik utama, yaitu rata-rata dan standar deviasi dari *pitch utterance*.

5) **Mengubah Representasi Kontinu Karakteristik Suara Menjadi Kategorikal**
Karakteristik suara kontinu yang telah diekstrak sebelumnya diubah menjadi representasi kategorikal dengan metode diskritisasi. Proses ini melibatkan pembagian rentang nilai kontinu menjadi tujuh interval yang disebut bin. Histogram dari distribusi nilai karakteristik digunakan untuk menentukan batas-batas bin secara optimal. Setiap nilai kontinu kemudian dipetakan ke bin yang sesuai, dan diberikan label kategori yang telah ditentukan sebelumnya. Contohnya, untuk karakteristik *speaking rate*, tujuh kategori yang mungkin adalah sangat lambat, agak lambat, sedikit lambat, kecepatan sedang, sedikit cepat, agak cepat, dan sangat cepat. Dengan representasi kategorikal ini, pengguna dapat dengan mudah memilih tingkat *speaking rate* yang diinginkan, sehingga mempermudah proses kontrol dan modifikasi output suara.

6) **Menciptakan Deskripsi untuk Setiap Audio berdasarkan Kategori Karakteristik Suara**
Masing-masing audio akan diberi deskripsi yang dihasilkan secara otomatis berdasarkan kategori karakteristik suaranya. Untuk mencapai hal ini, digunakan *Large Language Model* (LLM) *Gemma-2B* [16]. Kategori-kategori karakteristik suara akan diberikan sebagai input (prompt) kepada model, dan model akan menghasilkan teks deskriptif sebagai output. Sebagai contoh, jika inputnya adalah "*expressive voice, very slow speaking, slightly noisy*", maka output yang mungkin adalah "*In a very expressive voice, Jenny pronounces her words incredibly slowly. There's some noise in the recording*"

B. Proses Training

Proses training dilakukan pada notebook yang memiliki akses ke GPU berbasis CUDA untuk mempercepat proses pelatihan. Untuk melatih model, terlebih dahulu dilakukan pembagian dataset menjadi train dan test dengan rasio 8:2. Sebelum memulai pelatihan, akan dilakukan inisiasi *weight* pada model Parler-TTS yang sudah dilatih pada dataset berbahasa Inggris. Hal ini diperlukan agar model tidak perlu belajar dari awal dan dapat memanfaatkan pengetahuan yang telah didapat sebelumnya dalam mempelajari dataset yang baru. Setelah itu,

akan dilakukan pelatihan atau *fine-tuning* pada model Parler-TTS.

Hyperparameter tuning adalah salah satu cara yang dilakukan untuk meningkatkan performa model. *Learning rate* adalah *hyperparameter* yang mengatur seberapa besar langkah model dalam memperbarui bobotnya selama proses training. *Learning rate* yang terlalu besar dapat menyebabkan model tidak stabil dan sulit mencapai titik optimal, sedangkan learning rate yang terlalu kecil akan memperlambat proses training. Pemilihan learning rate yang tepat bertujuan untuk memastikan model belajar secara efektif dan stabil.

Jumlah *epoch* menentukan berapa kali seluruh dataset digunakan untuk melatih model. Jumlah *epoch* yang optimal dapat membantu model mempelajari data dengan baik dan efisien untuk menghindari *underfitting* maupun *overfitting*.

Batch size adalah jumlah sampel data yang diproses dalam satu iterasi sebelum model memperbarui bobotnya. Pengaturan *batch size* mempengaruhi bagaimana model mempelajari detail dan variasi dari data, namun *batch size* juga berpengaruh pada jumlah iterasi yang diperlukan untuk menyelesaikan satu *epoch*.

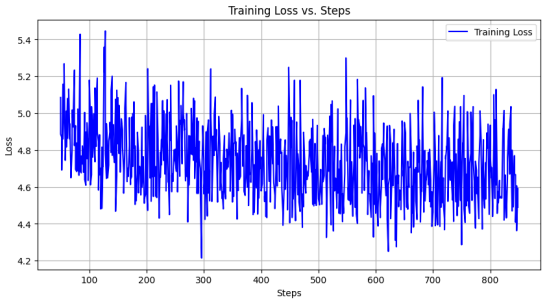
Gradient accumulation steps adalah metode untuk mengumpulkan gradien dari beberapa iterasi sebelum melakukan pembaruan bobot. Teknik ini memungkinkan simulasi *batch size* yang lebih besar tanpa membutuhkan memori tambahan, sehingga sangat berguna dalam lingkungan dengan keterbatasan sumber daya. Pengaturan langkah akumulasi gradien yang tepat dapat meningkatkan efisiensi training.

C. *Evaluasi*

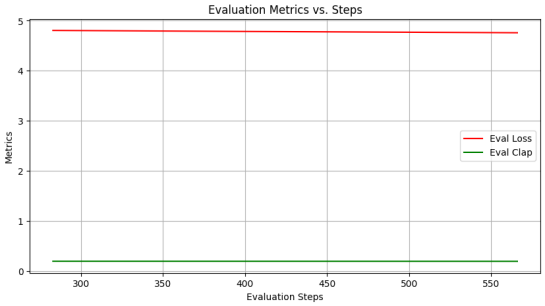
Evaluasi akan dilakukan menggunakan beberapa metrik, diantaranya Mean Opinion Score (MOS), CLAP similarity, dan WER metrics. MOS adalah metrik *crowdsourced* yang menilai hasil sintesis berdasarkan kejelasan, kealamian, keterpahaman, dan ekspresi. Metode evaluasi MOS merupakan metode evaluasi subjektif di mana pendengar diminta untuk memberikan skor pada kualitas audio yang dihasilkan oleh model TTS. CLAP similarity [17] adalah metrik yang digunakan untuk menggunakan kesesuaian antara deskripsi teks dan audio yang dihasilkan. Metrik ini menghitung skor kesamaan antara audio dan teks, di mana skor yang lebih tinggi menunjukkan audio yang dihasilkan lebih sesuai dengan deskripsi teks. Pada penelitian ini, digunakan model CLAP pada huggingface ”laion/larger-clap-music-and-speech”. Sementara itu, WER adalah metrik standar dalam pengenalan suara, tetapi juga relevan untuk mengevaluasi text-to-speech dalam konteks pemahaman dan reproduksi kata. Metrik ini mengukur kesalahan pengenalan kata dengan membandingkan transkrip yang dihasilkan model terhadap teks referensi. Pada penelitian ini, digunakan model Whisper [18] yang telah di-*fine tuning* pada bahasa Indonesia. Model yang digunakan adalah model huggingface ”cahya/whisper-medium-id”.

V. Hasil dan Analisis

Model yang telah dilatih dan dioptimasi dengan metode pelatihan yang telah dijelaskan sebelumnya kemudian dilakukan evaluasi. Untuk evaluasi yang bersifat objektif sendiri, model mendapatkan nilai CLAP *similarity* sebesar 0.21. Hasil ini masih belum terlalu baik dikarenakan *similarity* yang rendah dan juga cenderung lebih rendah dibandingkan percobaan yang dilakukan pada paper [1]. Untuk metrik WER, model mendapatkan nilai sebesar 51 persen. Hasil ini juga masih belum optimal. Namun, jika dilihat secara kualitatif dan subjektif, nilai *Mean Opinion Score* (MOS) untuk kejelasan audio (3.825) dan kemiripan dengan deskripsi (4.375) menunjukkan bahwa model mampu menghasilkan audio yang cukup dapat dipahami dan relevan dengan input deskripsi.



Gambar 2: Grafik *training model*



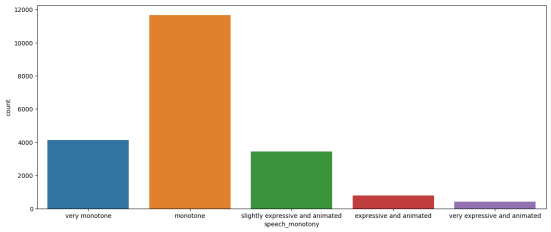
Gambar 3: Grafik *evaluation model*

Pada gambar 2 dan 3, terlihat sebenarnya model masih dalam proses pembelajaran. Hal ini terlihat dari tren *training loss* dan *validation* loss yang konsisten turun walaupun sedikit demi sedikit. Selain itu, hasil evaluasi *clap similarity* juga cenderung naik, yang menunjukkan audio yang dihasilkan semakin mirip dengan deskripsi, walaupun juga tidak dengan signifikan. Oleh karena itu, terdapat indikasi dimana proses training masih bisa diteruskan lebih lama lagi. Dengan ketersediaan sumber daya komputasi yang lebih memadai, potensi peningkatan kinerja model masih terbuka.

Jika dilihat berdasarkan WER lalu dianalisis secara kualitatif, masih terdapat beberapa audio yang pelafalannya kurang benar dan kurang jelas. Hal ini dapat disebabkan oleh beberapa penyebab diantaranya.

1) Terdapat beberapa transkrip yang tidak akurat, seperti masih terdapat beberapa kata yang *typo* atau salah ketik. Hal ini menyebabkan model kesusahan dalam melafalkan beberapa kata.

2) Kurangnya variasi suara pada dataset. Hal ini menyebabkan model menjadi kesusahan dalam meghasilkan audio yang memiliki karakteristik yang jarang muncul. Pada gambar 4 terlihat karakteristik suara yang ekspresif sangat sedikit jika dibandingkan dengan karakteristik suara lain. Hal ini membuat model kesulitan untuk menghasilkan suara dengan karakteristik suara seperti itu dengan jelas.
3) Transkrip pada dataset tidak memiliki tanda baca seperti koma, tanda tanya, ataupun titik. Hal ini menyebabkan model tidak belajar kapan suatu kalimat harus berhenti sejenak atau menggunakan nada yang sesuai.



Gambar 4: Persebaran karakteristik suara

VI. KESIMPULAN

Untuk saat ini, performa dari model parler-TTS menggunakan dataset berbahasa Indonesia masih tergolong kurang baik. Model masih perlu beradaptasi dengan data berbahasa indonesia terlebih dahulu. Penggunaan dataset yang berkualitas dapat membantu meningkatkan kualitas audio yang dihasilkan. Selain itu, penambahan lebih banyak data ataupun penambahan iterasi pelatihan juga dapat menjadi salah satu pendekatan untuk menciptakan model yang lebih baik dan terbiasa dengan bahasa Indonesia.

REFERENSI

[1] D. Lyth and S. King, "Natural language guidance of high-fidelity text-to-speech with synthetic annotations," 2024. [Online]. Available: https://arxiv.org/abs/2402.01912

[2] Y. Wang, D. Stanton, Y. Zhang, R. Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, F. Ren, Y. Jia, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," 2018. [Online]. Available: https://arxiv.org/abs/1803.09017

[3] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. J. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron," 2018. [Online]. Available: https://arxiv.org/abs/1803.09047

[4] A. Vyas, B. Shi, M. Le, A. Tjandra, Y.-C. Wu, B. Guo, J. Zhang, X. Zhang, R. Adkins, W. Ngan, J. Wang, I. Cruz, B. Akula, A. Akinyemi, B. Ellis, R. Moritz, Y. Yungster, A. Rakotoarison, L. Tan, C. Summers, C. Wood, J. Lane, M. Williamson, and W.-N. Hsu, "Audiobox: Unified audio generation with natural language prompts," 2023. [Online]. Available: https://arxiv.org/abs/2312.15821

[5] Y. Leng, Z. Guo, K. Shen, X. Tan, Z. Ju, Y. Liu, Y. Liu, D. Yang, L. Zhang, K. Song, L. He, X.-Y. Li, S. Zhao, T. Qin, and J. Bian, "Prompttts 2: Describing and generating voices with text prompt," 2023. [Online]. Available: https://arxiv.org/abs/2309.02285

[6] J. Jung, A. Jansson, and D. Jeong, "Musicgen-chord: Advancing music generation through chord progressions and interactive web-ui," 2024. [Online]. Available: https://arxiv.org/abs/2412.00325

[7] M. Kawamura, R. Yamamoto, Y. Shirahata, T. Hasumi, and K. Tachibana, "Libritts-p: A corpus with speaking style and speaker identity prompts for text-to-speech and style captioning," 2024. [Online]. Available: https://arxiv.org/abs/2406.07969

[8] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," 2024. [Online]. Available: https://arxiv.org/abs/2306.05284

[9] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, "Scaling instruction-finetuned language models," 2022. [Online]. Available: https://arxiv.org/abs/2210.11416

[10] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, "High-fidelity audio compression with improved rvqgan," 2023. [Online]. Available: https://arxiv.org/abs/2306.06546

[11] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," 2022. [Online]. Available: https://arxiv.org/abs/2210.13438

[12] W. Wongso, A. Joyoadikusumo, and S. Limcorn, "g2p id," https://github.com/bookbot-kids/g2p_id, 2024.

[13] M. Lavechin, M. Métais, H. Titeux, A. Boissonnet, J. Copet, M. Rivière, E. Bergelson, A. Cristia, E. Dupoux, and H. Bredin, "Brouhaha: multi-task training for voice activity detection, speech-to-noise ratio, and c50 room acoustics estimation," 2023. [Online]. Available: https://arxiv.org/abs/2210.13248

[14] A. Kumar, K. Tan, Z. Ni, P. Manocha, X. Zhang, E. Henderson, and B. Xu, "Torchaudio-squim: Reference-less speech quality and intelligibility measures in torchaudio," 2023. [Online]. Available: https://arxiv.org/abs/2304.01448

[15] M. Morrison, C. Hsieh, N. Pruyne, and B. Pardo, "Cross-domain neural pitch and periodicity estimation," 2024. [Online]. Available: https://arxiv.org/abs/2301.12258

[16] G. Team, "Gemma," 2024. [Online]. Available: https://www.kaggle.com/m/3301

[17] B. Elizalde, S. Deshmukh, M. A. Ismail, and H. Wang, "Clap: Learning audio concepts from natural language supervision," 2022. [Online]. Available: https://arxiv.org/abs/2206.04769

[18] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022. [Online]. Available: https://arxiv.org/abs/2212.04356