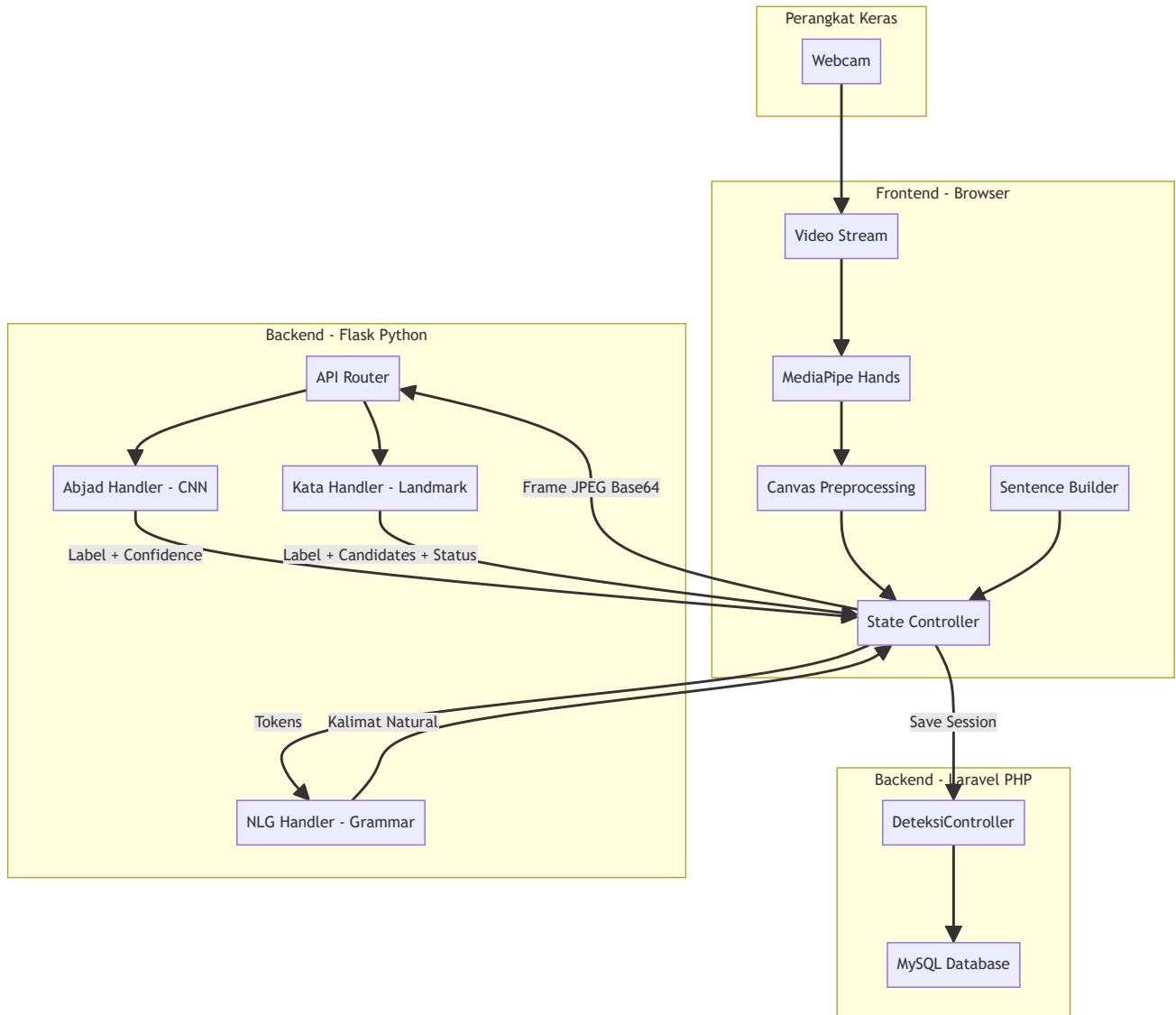


# Dokumentasi Arsitektur Mode Kalimat

Dokumen teknis lengkap arsitektur **Mode Kalimat** pada sistem Deteksi BISINDO. Disusun untuk kebutuhan skripsi.

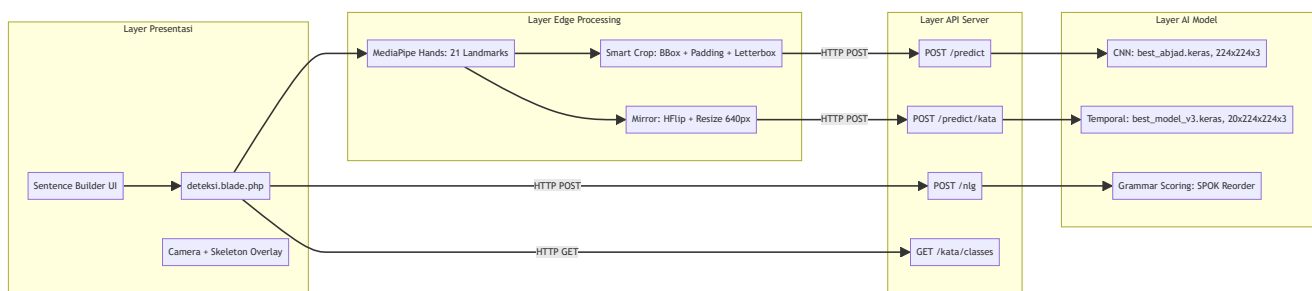
## 1. Diagram Blok Sistem



### Penjelasan Komponen:

Komponen	Teknologi	Peran
Webcam	getUserMedia API	Menangkap video stream real-time
MediaPipe Hands	TensorFlow.js	Melacak 21 titik landmark tangan di browser
Canvas Preprocessing	HTML5 Canvas API	Smart Crop, Mirroring, konversi Base64
State Controller	JavaScript	Logika switching pipeline Abjad dan Kata
Sentence Builder	HTML + JavaScript	UI penyusun kalimat dari kata dan ejaan
DeteksiController	Laravel PHP	Halaman dan penyimpanan sesi latihan
API Router	Flask Python	Distribusi request ke handler yang sesuai
Abjad Handler	TensorFlow CNN	Klasifikasi gambar tangan menjadi huruf A-Z
Kata Handler	TF + MediaPipe Holistic	Klasifikasi sekuens gerakan menjadi kata
NLG Handler	Python Rule-based	Transformasi token menjadi kalimat natural
MySQL	Database Relasional	Penyimpanan data sesi latihan

## 2. Arsitektur Perangkat Lunak

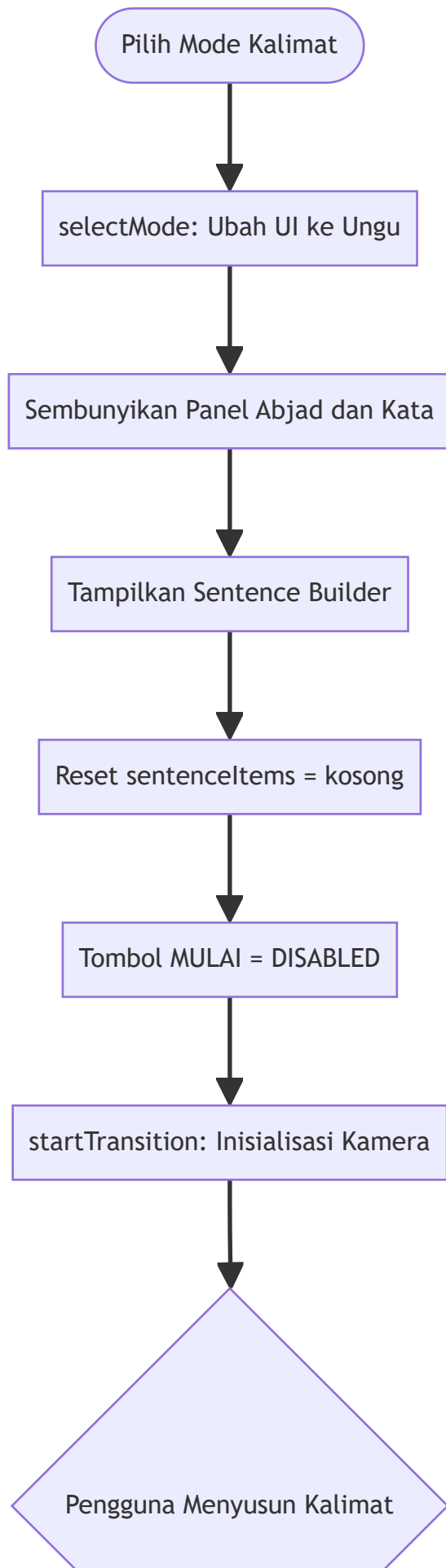


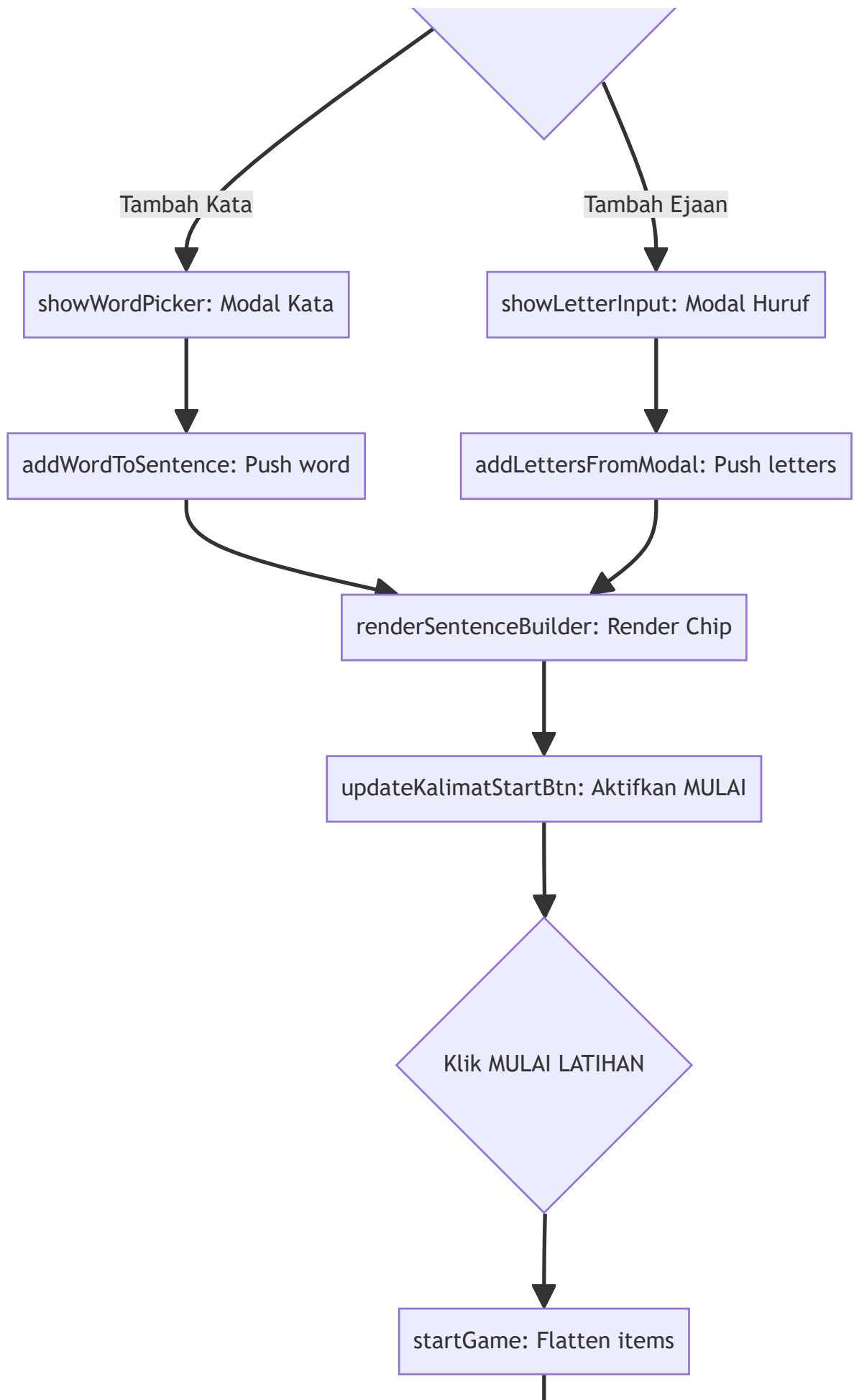
### Penjelasan Layer:

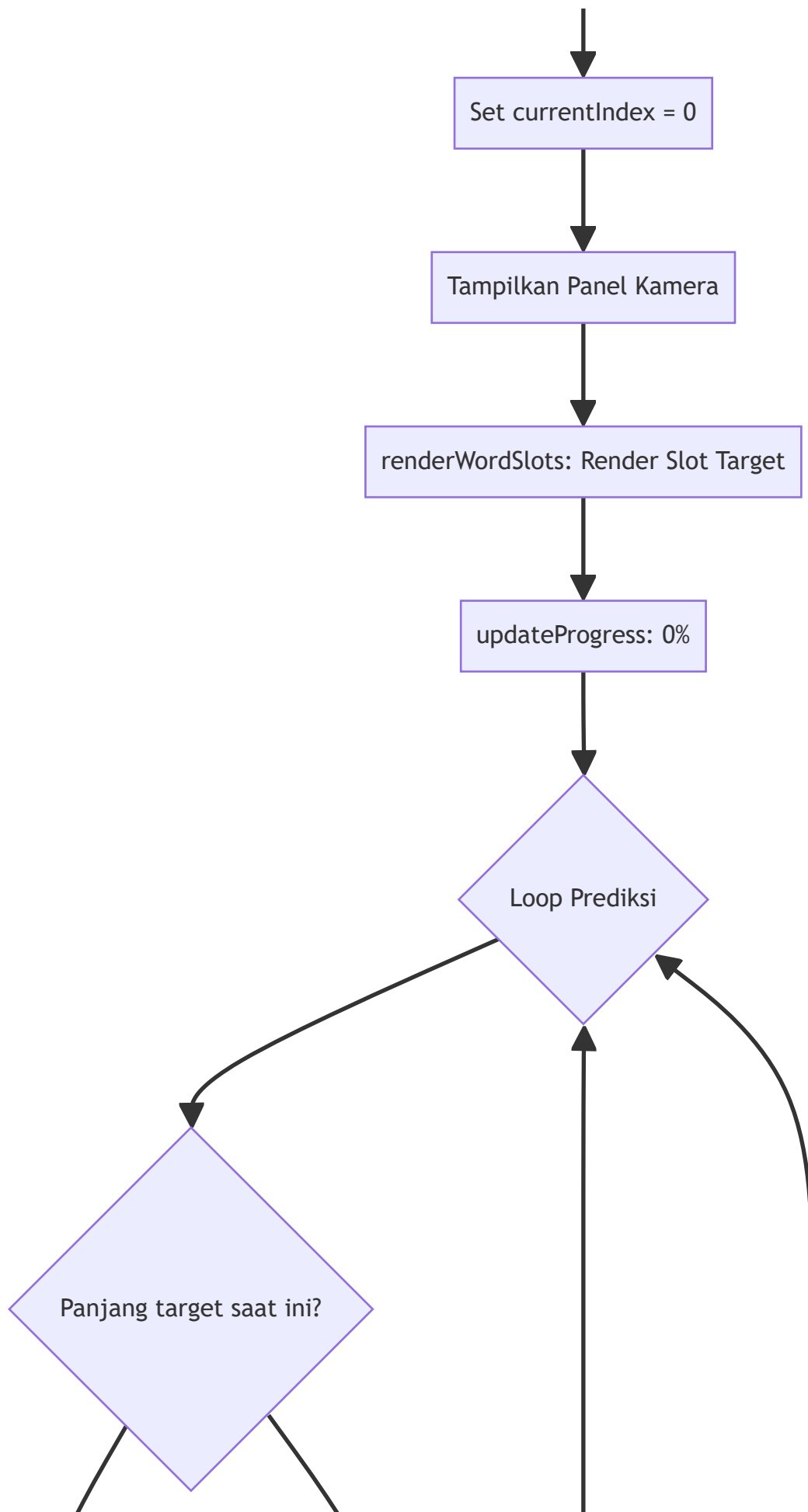
- **Layer Presentasi:** File `deteksi.blade.php` (2252 baris) berisi antarmuka pengguna, Sentence Builder, panel kamera, dan modal penyelesaian.
- **Layer Edge Processing:** Preprocessing gambar di browser sebelum dikirim ke server, mengurangi beban jaringan.
- **Layer API:** Flask menerima HTTP request dan mendistribusikan ke handler berdasarkan endpoint.
- **Layer AI Model:** Model TensorFlow/Keras yang telah dilatih melakukan inferensi. Grammar engine memproses token menjadi teks natural.

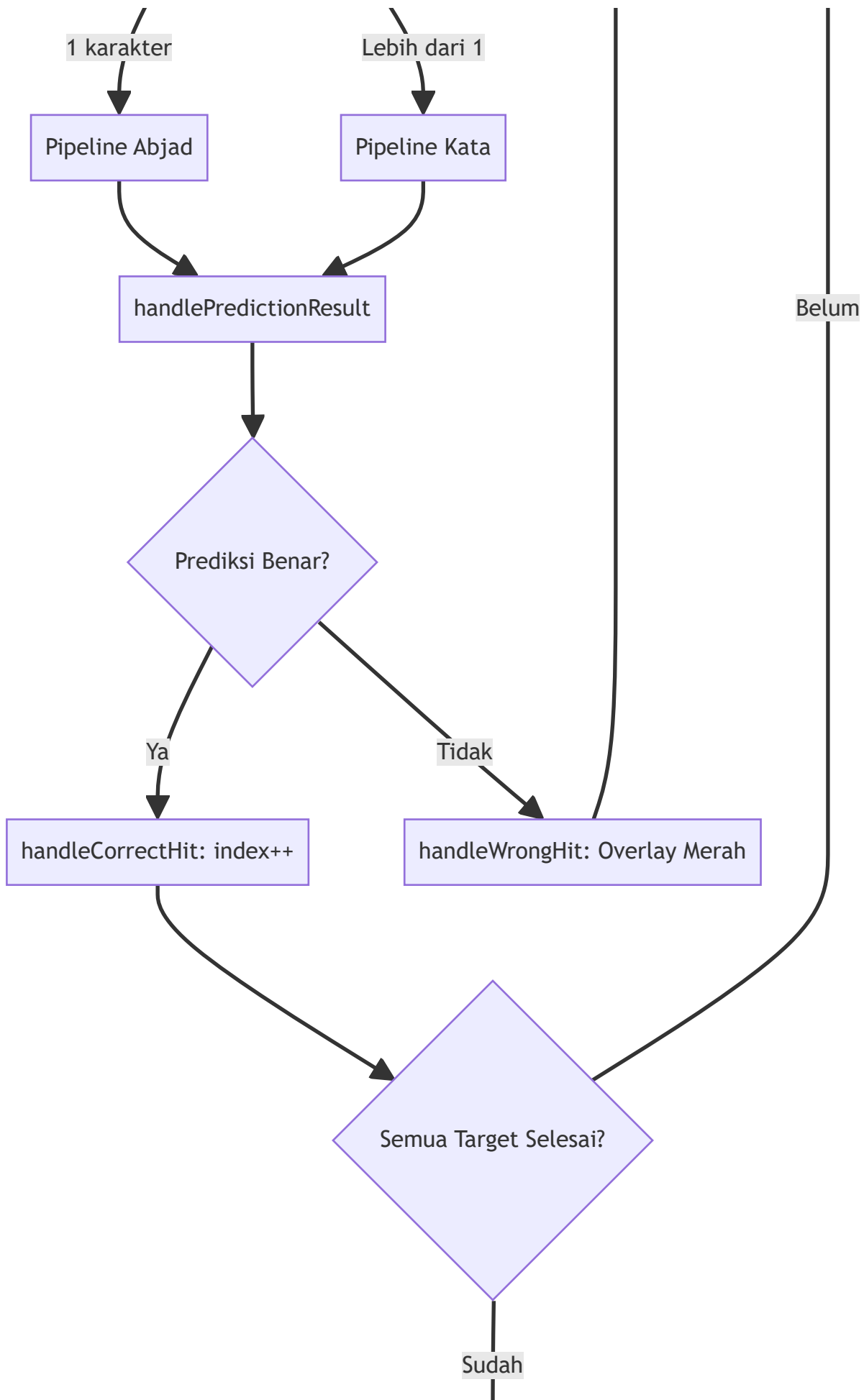
### 3. Flowchart Alur Utama

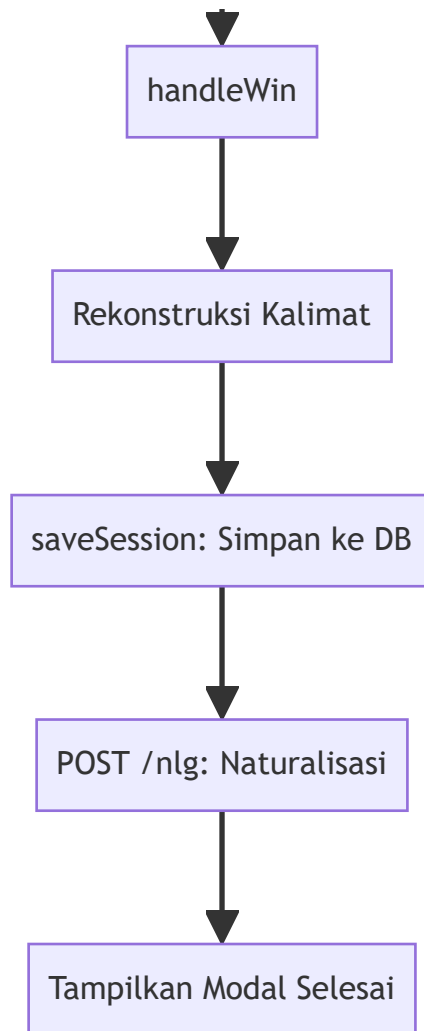
---







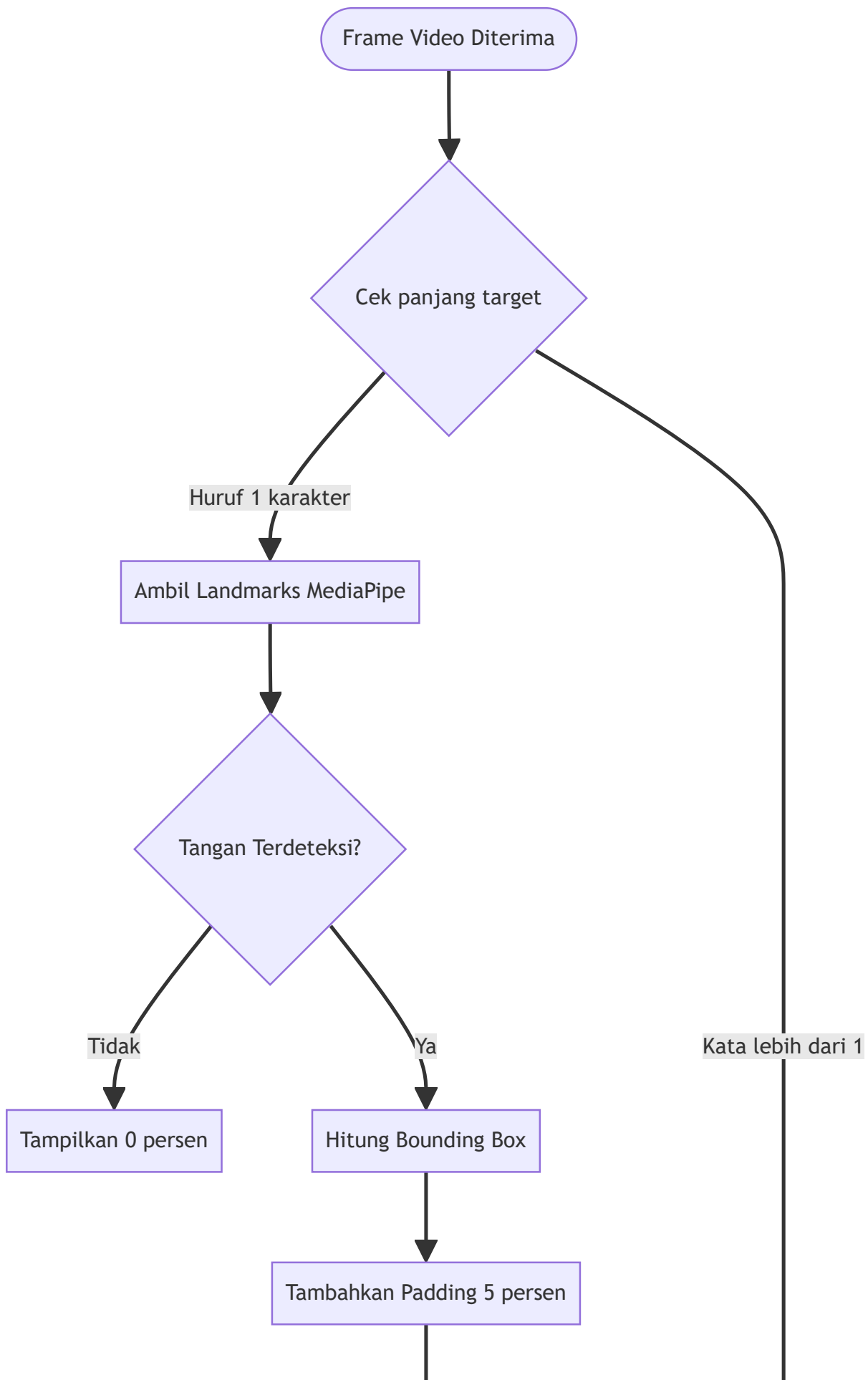


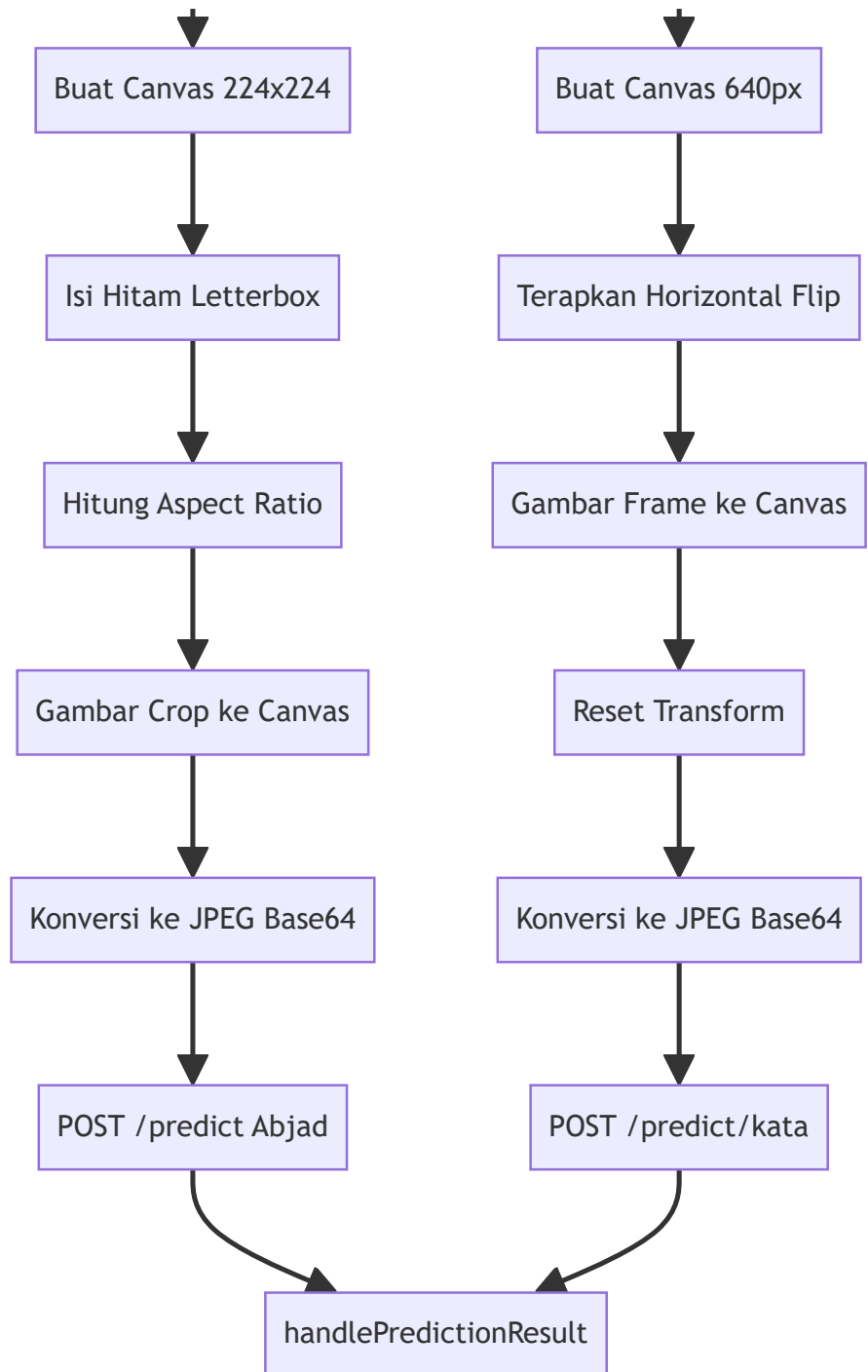




## 4. Flowchart Preprocessing Detail

---



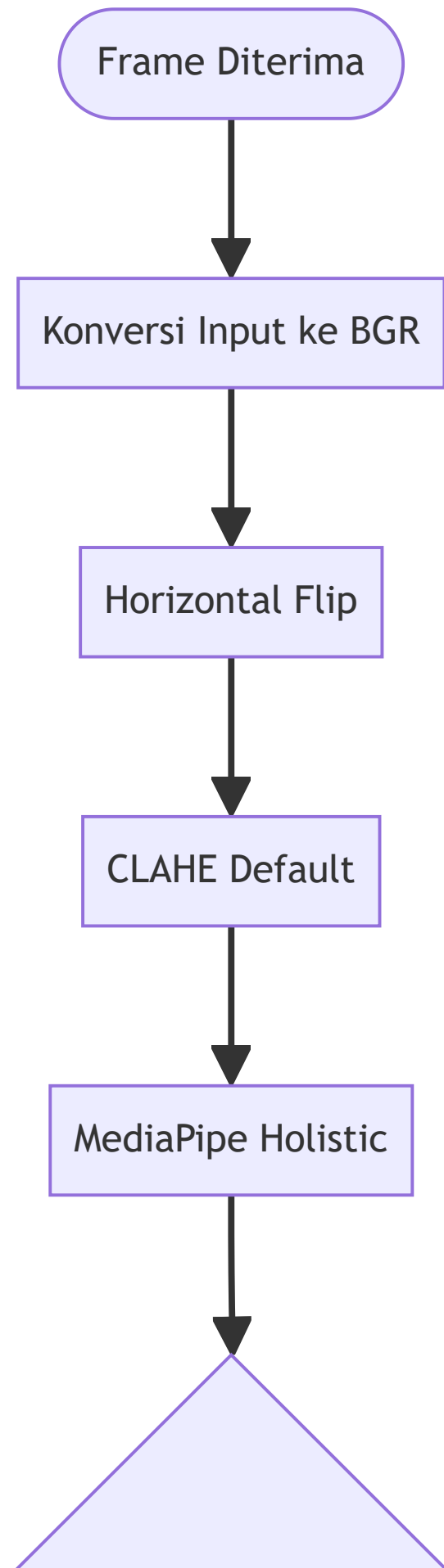


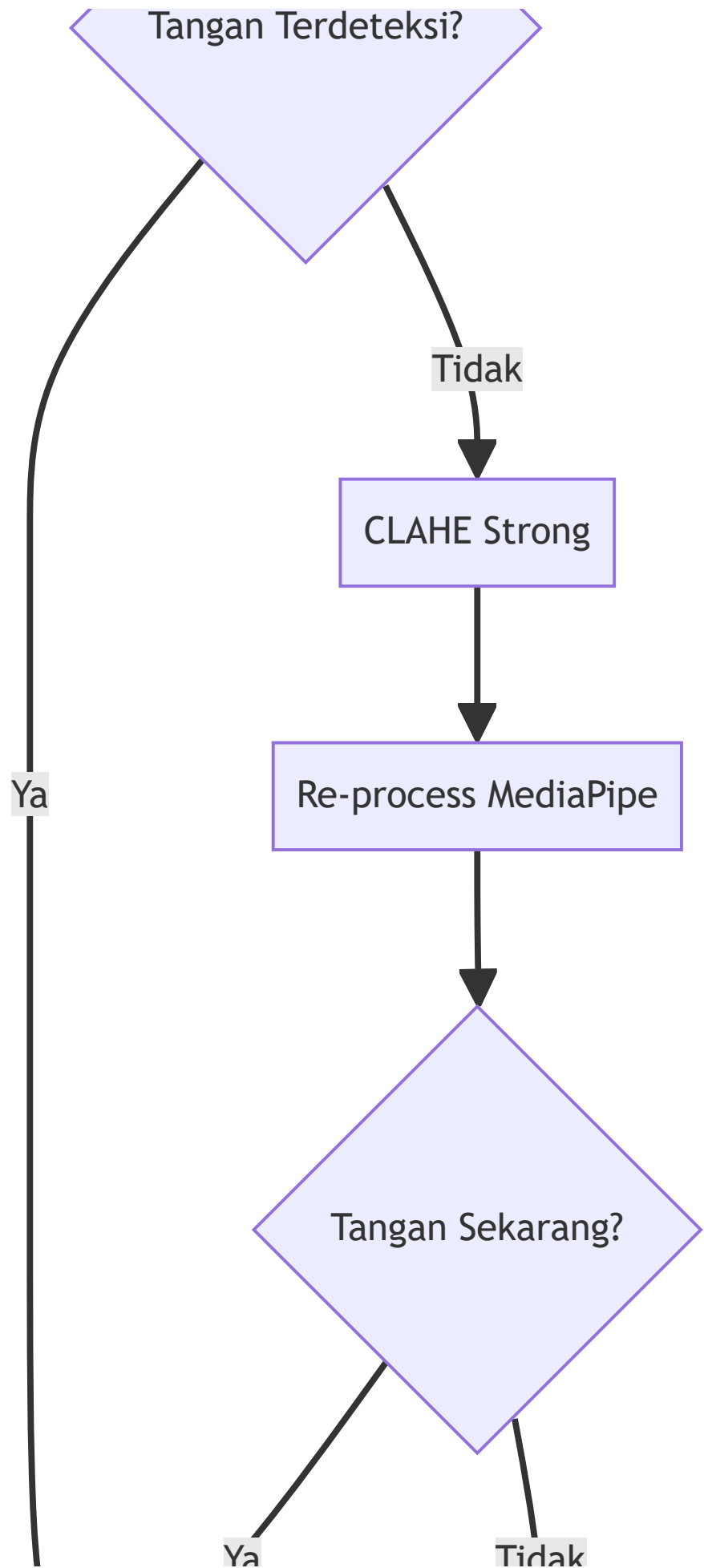
**Penjelasan Pipeline Abjad:** - MediaPipe frontend menyediakan koordinat 21 titik landmark tangan. - Bounding box dihitung dari koordinat min/max landmark. - Padding 5% ditambahkan di setiap sisi agar gesture tidak terpotong. - Canvas 224x224 diisi hitam terlebih dahulu (letterboxing). - Crop digambar ke canvas dengan aspect ratio terjaga untuk menghindari distorsi. - Hasil dikonversi ke JPEG Base64 quality 80% dan dikirim ke API Abjad.

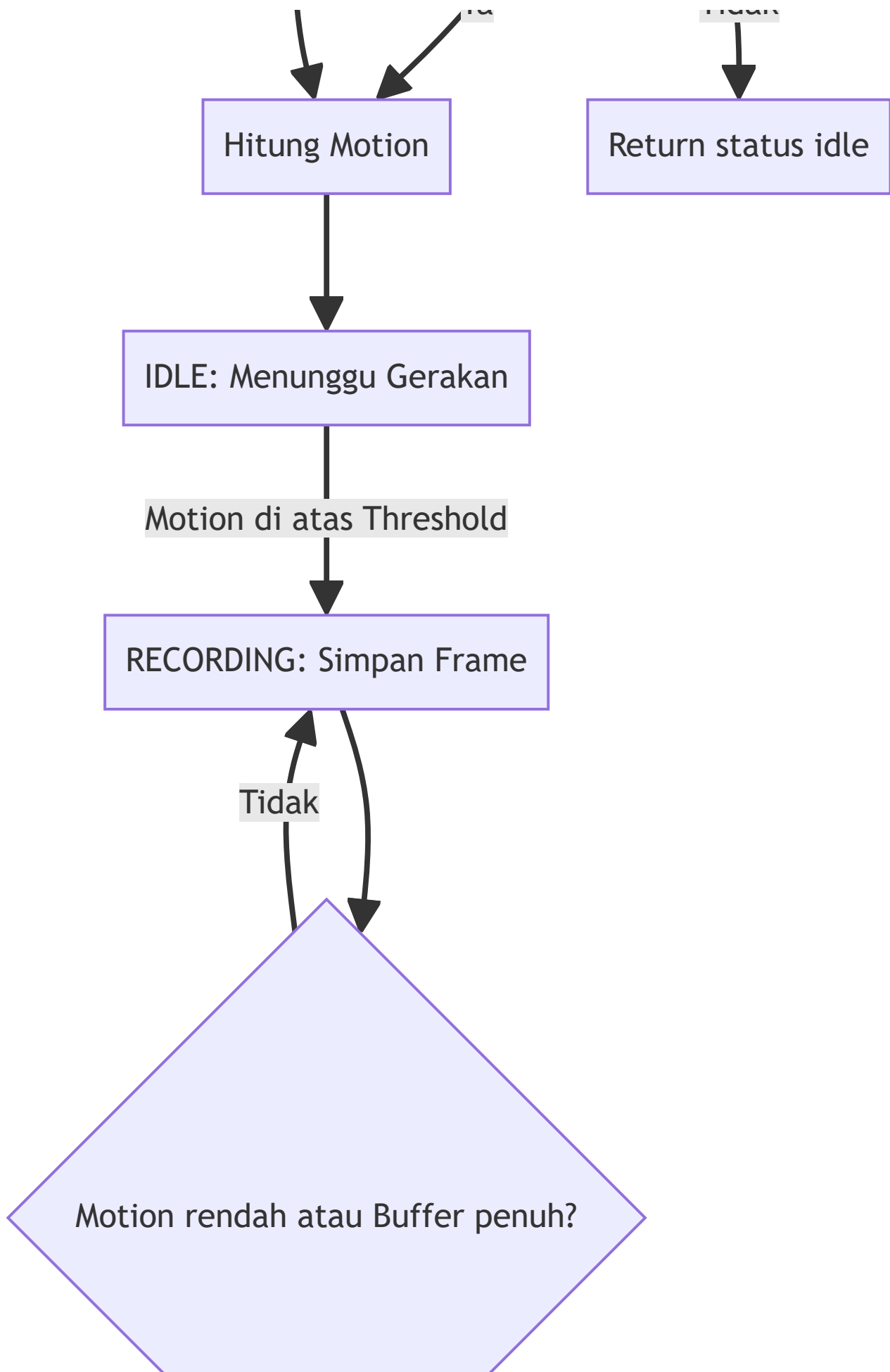
**Penjelasan Pipeline Kata:** - Canvas baru dibuat dengan lebar 640 piksel (tinggi proporsional). - Frame di-flip horizontal menggunakan `translate(width,0)` dan `scale(-1,1)` agar sesuai orientasi data latih (mirror). - Hasil dikirim ke API Kata yang menjalankan State Machine di backend.

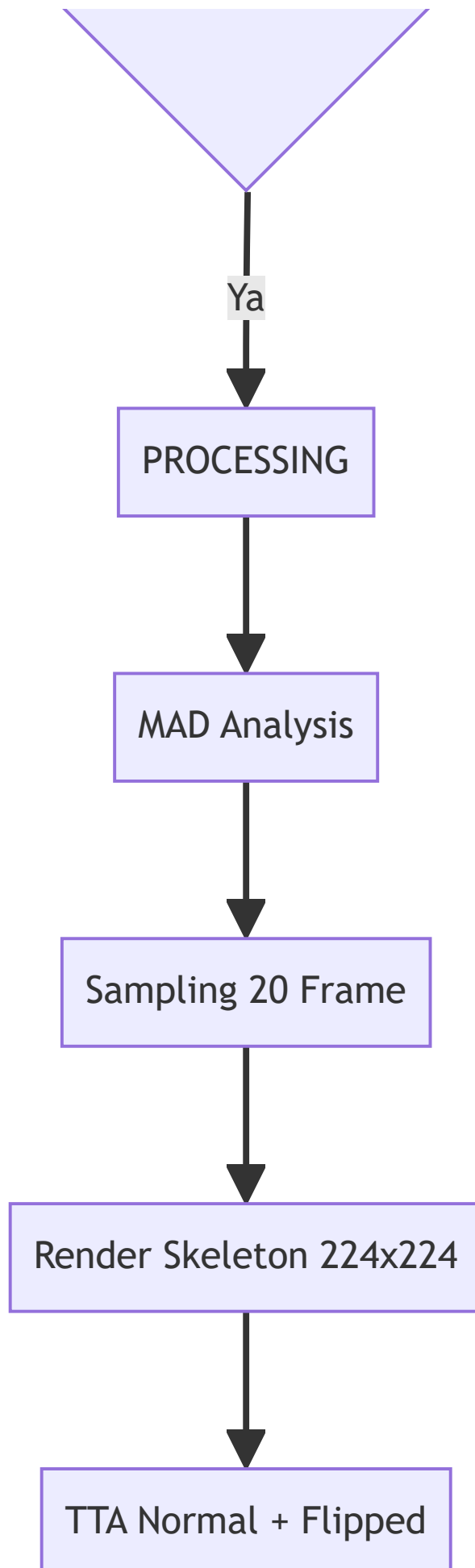
## 5. Flowchart State Machine Kata Handler

---

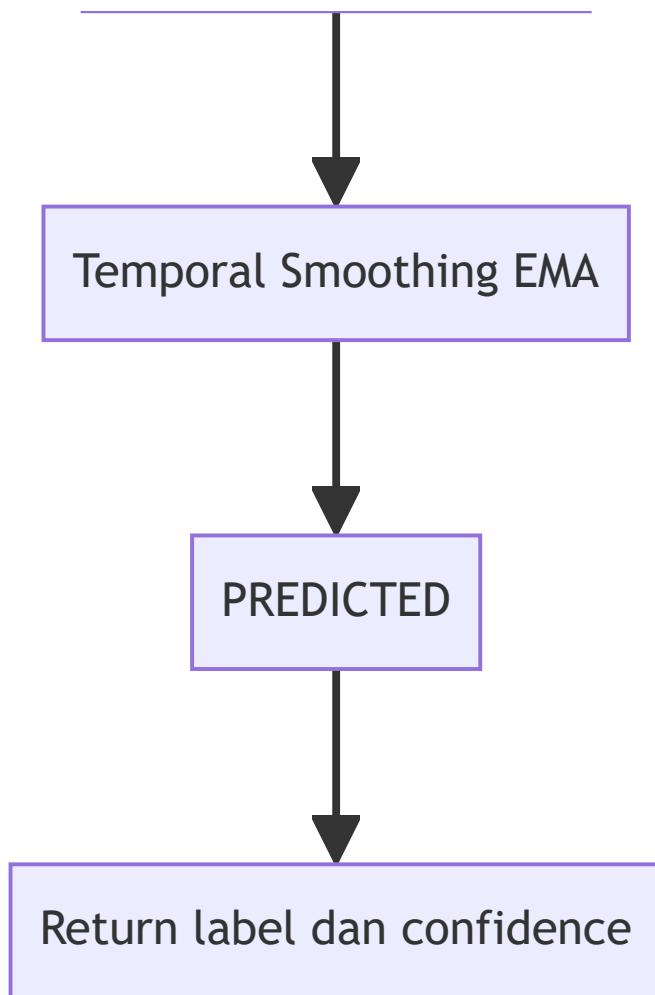












#### Penjelasan Detail Setiap State:

**IDLE** — Sistem menunggu tangan muncul dan gerakan signifikan terdeteksi. Dua metode deteksi gerakan digunakan secara paralel: (a) Skeleton-based: menghitung perpindahan pusat tangan antar-frame menggunakan landmark MediaPipe, threshold mulai = 1.5; (b) Pixel-based: menghitung perbedaan piksel langsung tanpa lag, threshold mulai = 1.2. Jika salah satu melampaui threshold, status berpindah ke RECORDING.

**RECORDING** — Setiap frame disimpan ke buffer bersama hasil MediaPipe Holistic. Pre-buffer menyimpan 10 frame sebelum gerakan dimulai agar awal gesture tidak hilang. Sistem memonitor gerakan secara kontinu. Jika gerakan turun di bawah stop threshold (skeleton: 1.0, pixel: 0.6) selama 3 frame berturut-turut (STOP\_PATIENCE), atau buffer mencapai batas 120 frame, status berpindah ke PROCESSING.

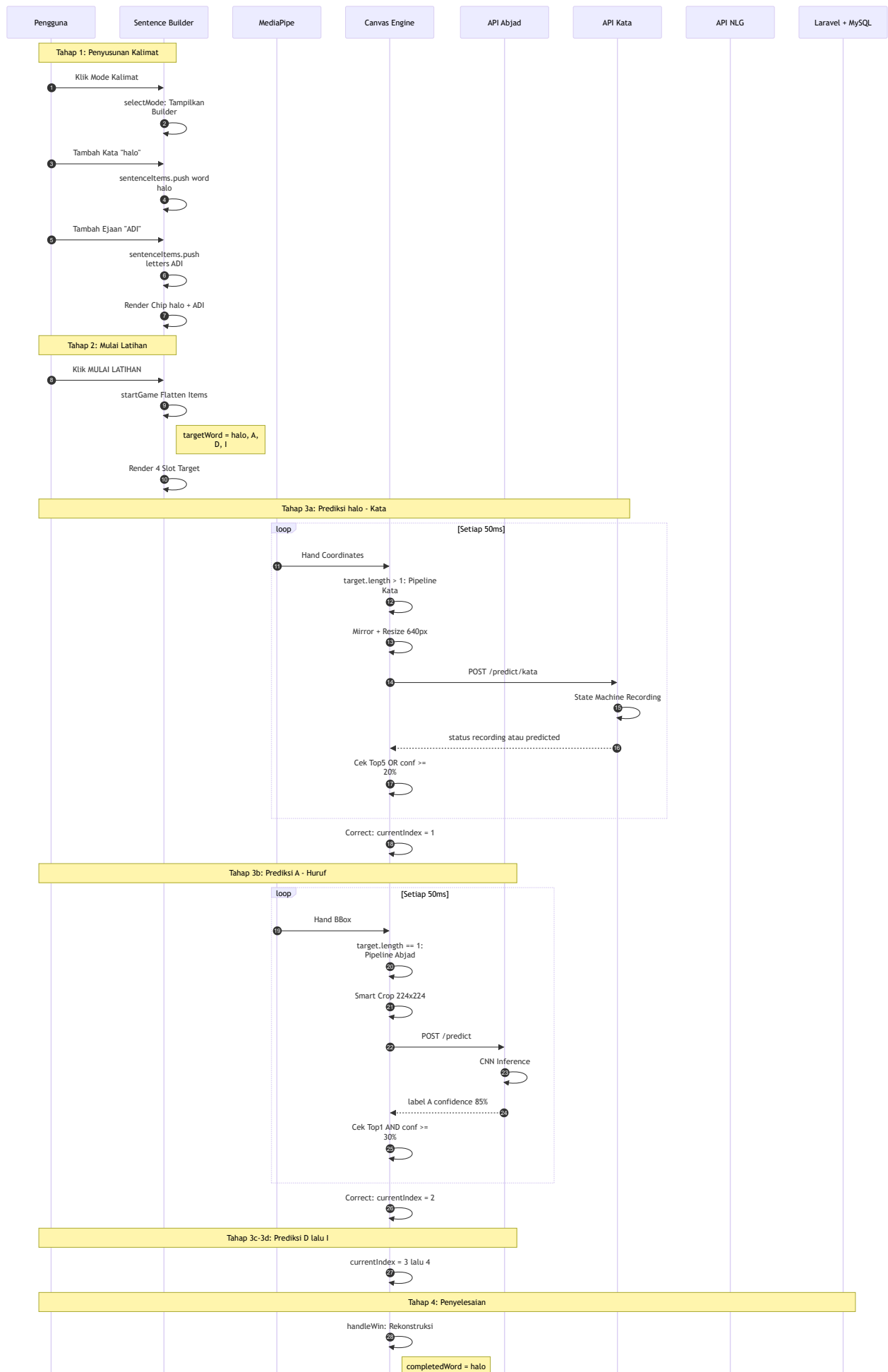
**PROCESSING** — Fungsi `trim_active_segment_by_motion()` menganalisis buffer menggunakan MAD (Median Absolute Deviation). Frame dikonversi ke grayscale 160x160, dihitung perbedaan absolut antar-frame, dihaluskan dengan kernel rata-rata  $k=5$ . Median dan MAD dari sinyal gerakan dihitung. Frame dengan gerakan di atas threshold ( $\text{median} + \text{MAD} \times 1.0$ ) dianggap aktif. Hanya segmen aktif ditambah margin 2 frame yang digunakan. Dari segmen aktif, 20 frame di-sample secara uniform. Setiap frame diproses dengan `render_v3()` yang menggambar skeleton (pose + tangan) ke canvas 224x224 dengan color-coding: hijau untuk tangan kiri, merah untuk tangan kanan, biru untuk pose. Landmark Persistence diterapkan: jika MediaPipe gagal deteksi tangan pada satu frame, posisi terakhir digunakan sebagai pengganti.

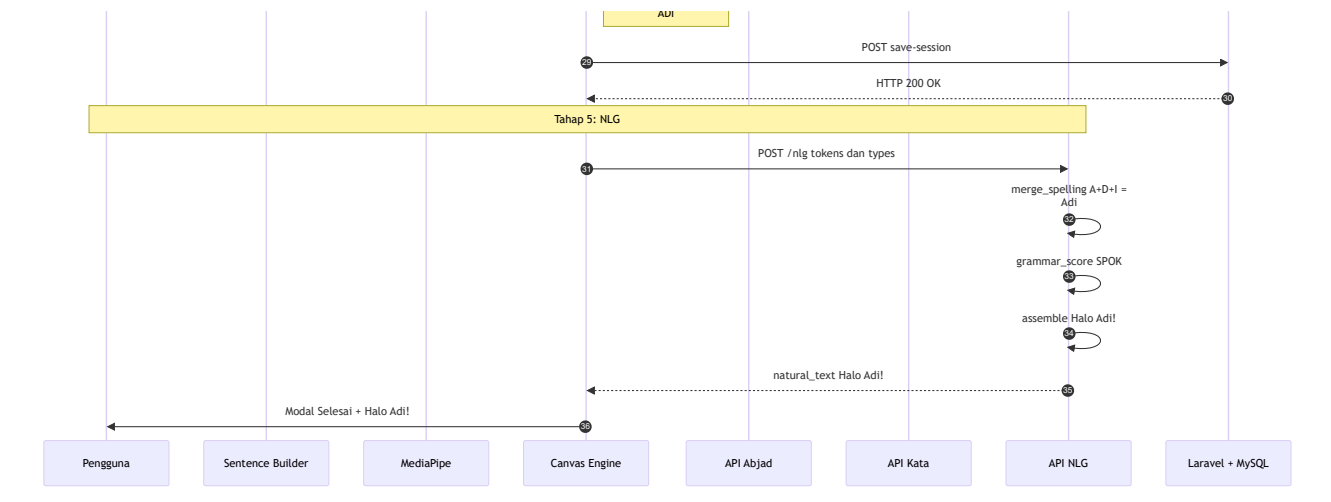
**TTA (Test-Time Augmentation)** — Model dijalankan 2 kali: sekali dengan orientasi normal, sekali dengan orientasi terbalik horizontal. Kedua hasil probabilitas dirata-ratakan untuk meningkatkan robustness terhadap variasi tangan kiri/kanan.

**Temporal Smoothing** — Probabilitas prediksi dihaluskan menggunakan EMA dari 3 prediksi terakhir dengan bobot [0.2, 0.3, 0.5]. Prediksi terbaru diberi bobot terbesar agar tetap responsif namun stabil.

## 6. Sequence Diagram Interaksi Lengkap

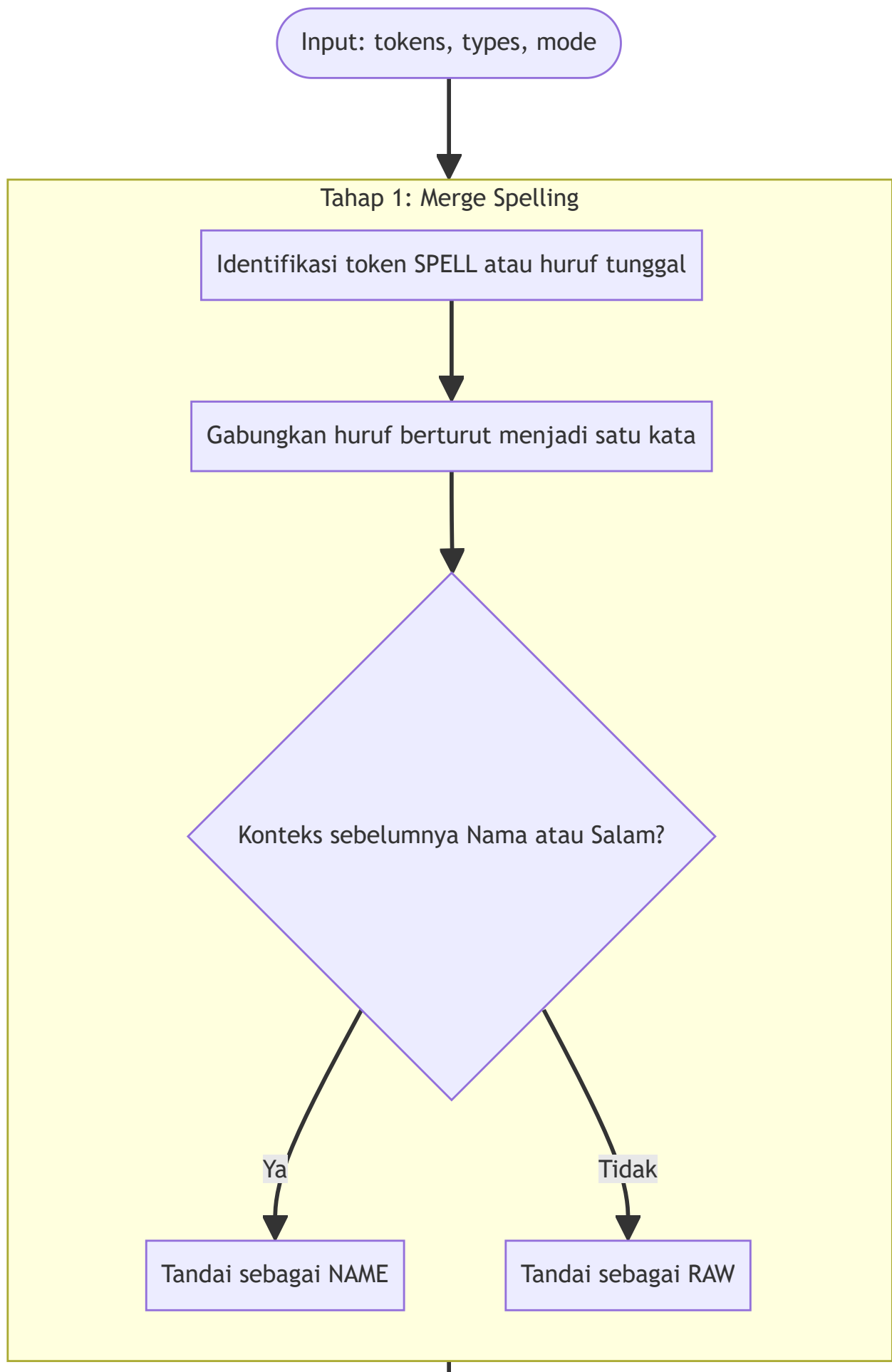
---

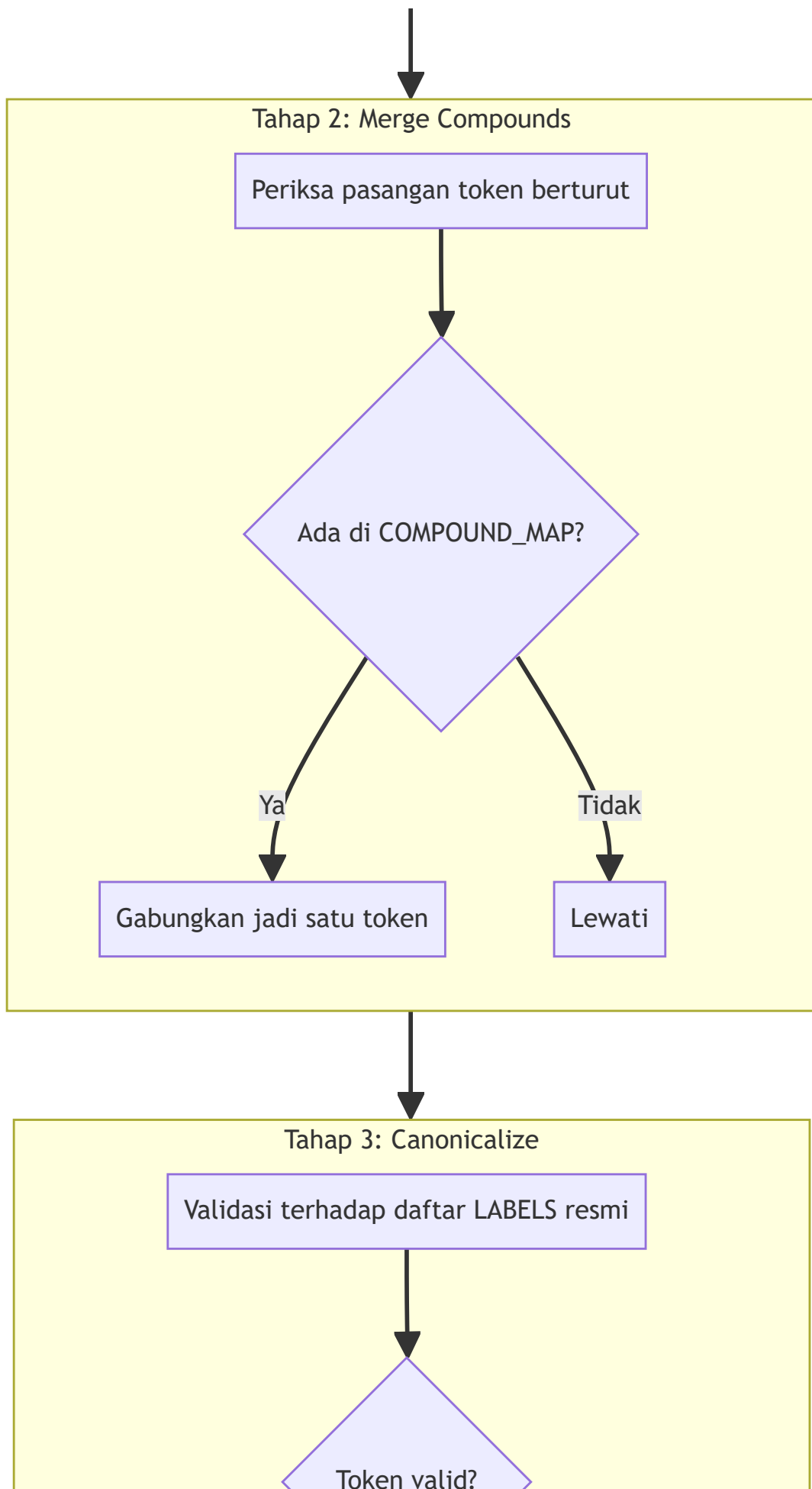


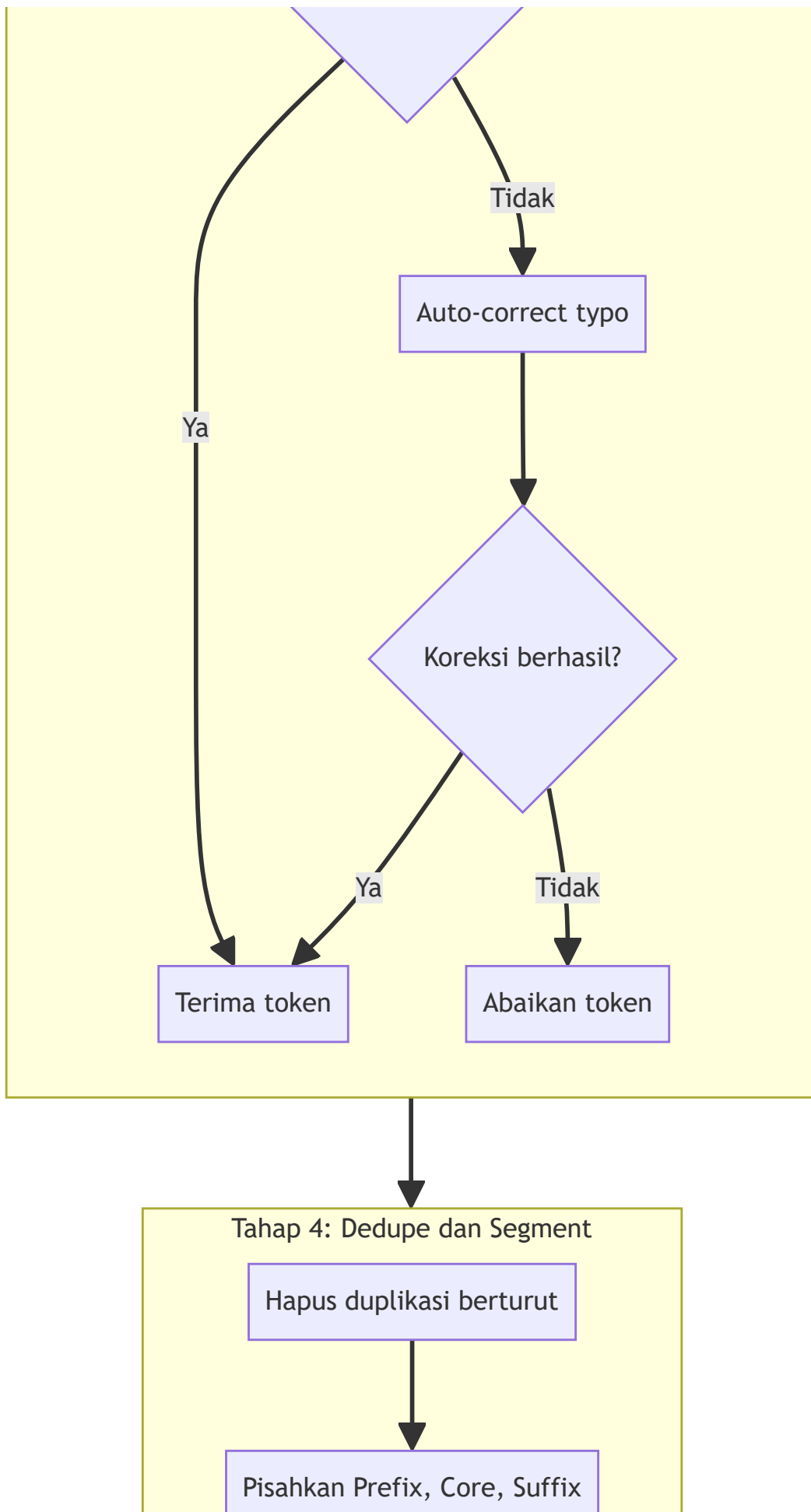


## 7. Flowchart Pipeline NLG

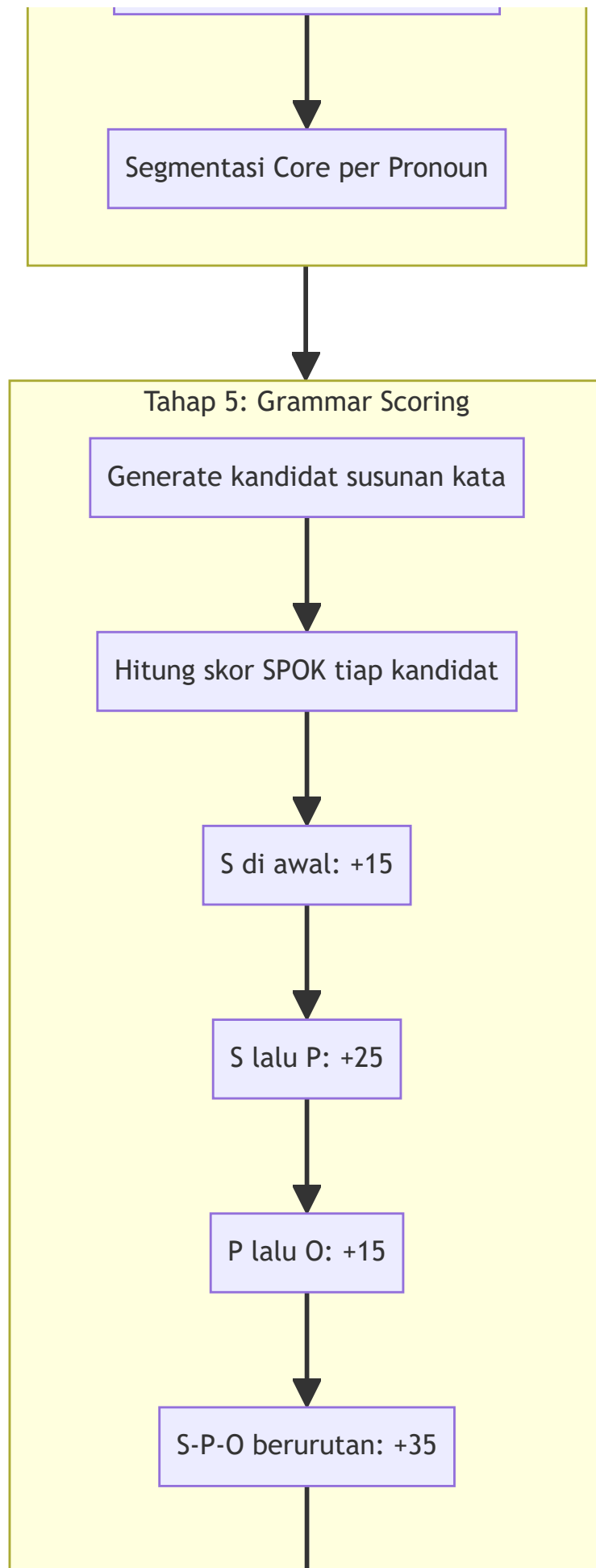
---

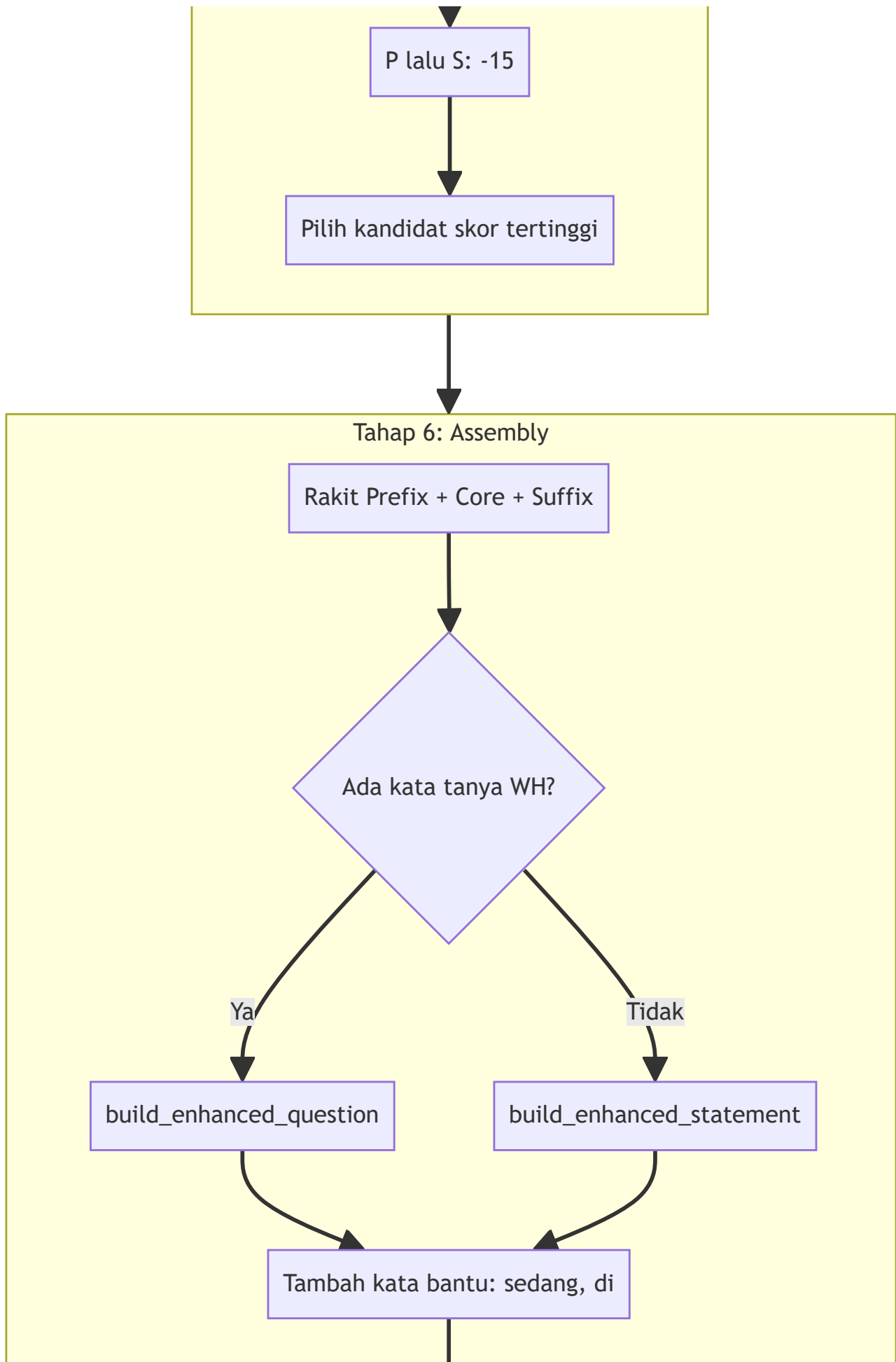


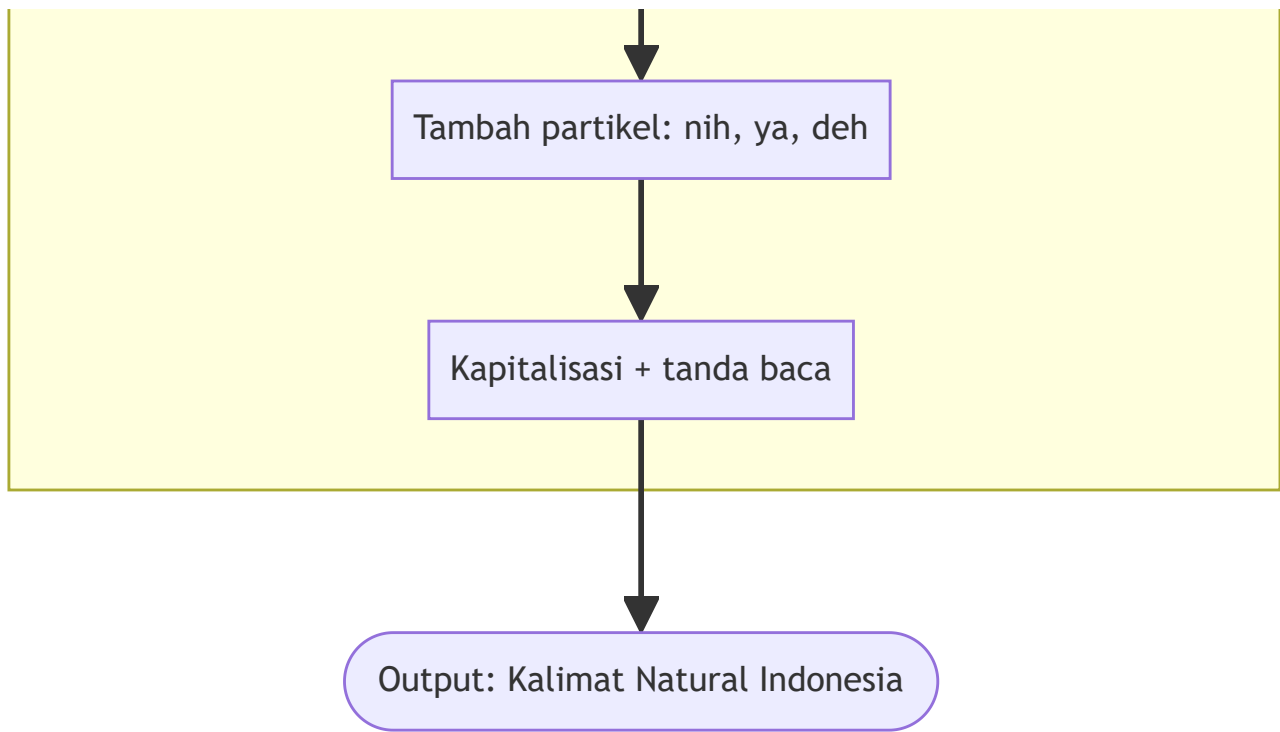










**Contoh Transformasi NLG:**

Input Tokens	Proses	Output Natural
[halo, A, D, I]	A+D+I merged sebagai NAME:Adi	"Halo, Adii"
[saya, makan, nasi]	SPOK sudah natural	"Saya sedang makan nasi."
[kamu, belajar, apa]	WH token terdeteksi, reorder	"Kamu sedang belajar apa?"
[selamat, pagi, terima kasih]	Compound merge + greeting variation	"Selamat pagi! Terima kasih!"

## 8. Tabel Parameter Eksperimental

No	Parameter	Variabel	Nilai	Keterangan
1	Dimensi Input CNN	IMG_SIZE	224x224x3	Resolusi setelah Smart Crop
2	Panjang Sekuens	T	20 frame	Frame di-sample dari buffer
3	Dimensi Render Skeleton	OUT_SIZE	224x224x3	Canvas skeleton
4	Lebar Frame Kata	targetW	640 px	Resolusi kirim ke server
5	Confidence Abjad	REQUIRED_CONFIDENCE	30%	Threshold minimum huruf
6	Confidence Kata	REQUIRED_CONFIDENCE_KATA	20%	Threshold minimum kata
7	Motion Start Skeleton	MOTION_START_THRESHOLD	1.5	Mulai rekaman
8	Motion Stop Skeleton	MOTION_STOP_THRESHOLD	1.0	Hentikan rekaman
9	Motion Start Pixel	PIXEL_MOTION_START_THRESHOLD	1.2	Deteksi piksel mulai
10	Motion Stop Pixel	PIXEL_MOTION_STOP_THRESHOLD	0.6	Deteksi piksel berhenti
11	Stop Patience	STOP_PATIENCE	3 frame	Frame diam sebelum stop
12	Min Gesture Frames	MIN_GESTURE_FRAMES	5 frame	Minimum gesture valid
13	Max Buffer Frames	MAX_BUFFER_FRAMES	120 frame	Kapasitas buffer
14	Pre-buffer Size	pre_buffer	10 frame	Frame sebelum gerakan
15	Prediction Cooldown	PREDICTION_COOLDOWN	0.5 detik	Jarak antar prediksi
16	Frame Skip Rate	FRAME_SKIP_RATE	2	Proses tiap frame ke-N
17	CLAHE Default	clipLimit	2.0	Equalization normal
18	CLAHE Strong	clipLimit	4.0	Equalization gelap
19	EMA Weights	smoothing	0.2, 0.3, 0.5	Bobot temporal
20	Smart Crop Padding	padding	5%	Margin bounding box
21	JPEG Quality	toDataURL	80%	Kualitas kompresi
22	Skeleton Scale	max_dim	1.3x	Faktor zoom skeleton
23	MAD Factor	MOTION_THRESHOLD_FACTOR	1.0	Pengali MAD
24	Min Hand Frames	MIN_HAND_FRAMES	3 frame	Min frame dgn tangan
25	Throttle Delay	setTimeout	50 ms	Jeda antar-request

## 9. Pengujian BLEU Score - Evaluasi NLG

### 9.1 Metodologi

BLEU (Bilingual Evaluation Understudy) digunakan untuk mengukur kualitas output NLG Mode Kalimat dengan membandingkan kalimat yang dihasilkan sistem terhadap referensi kalimat natural yang ditulis secara manual. Evaluasi menggunakan library `nltk.translate.bleu_score` dengan **Smoothing Method 1** untuk menangani kalimat pendek.

**Metrik yang diukur:** - **BLEU-1 (Unigram Precision):** Mengukur kecocokan kata per kata antara output dan referensi. Bobot: (1, 0, 0, 0). - **BLEU-2 (Bigram Precision):** Mengukur kecocokan pasangan kata berturut-turut. Bobot: (0.5, 0.5, 0, 0).

**Tokenisasi:** Teks di-lowercase, tanda baca dihapus, lalu dipecah berdasarkan spasi.

### 9.2 Kategori Pengujian

Evaluasi dilakukan terhadap **81 test case** yang mencakup **seluruh 26 label kata BISINDO**, dibagi dalam 3 kategori:

No	Kategori	Jumlah Test Case	Deskripsi
A	Kalimat Terstruktur	30	Input token dalam urutan SPOK yang benar
B	Kalimat Acak	25	Input token dalam urutan terbalik/acak, menguji kemampuan reordering SPOK
C	Kata Tunggal	26	Setiap label kata diuji secara individual

**Cakupan Label:** Semua 26 label tercakup — Apa, Apa Kabar, Bagaimana, Baik, Belajar, Berapa, Bingung, Dia, Halo, Kamu, Makan, Marah, Melihat, Menulis, Sabar, Saya, Selamat Malam, Selamat Pagi, Selamat Siang, Selamat Sore, Siapa, Terima Kasih, Tidur, Tinggi, Tuli, Maaf.

### 9.3 Hasil Evaluasi

#### Ringkasan Per Kategori:

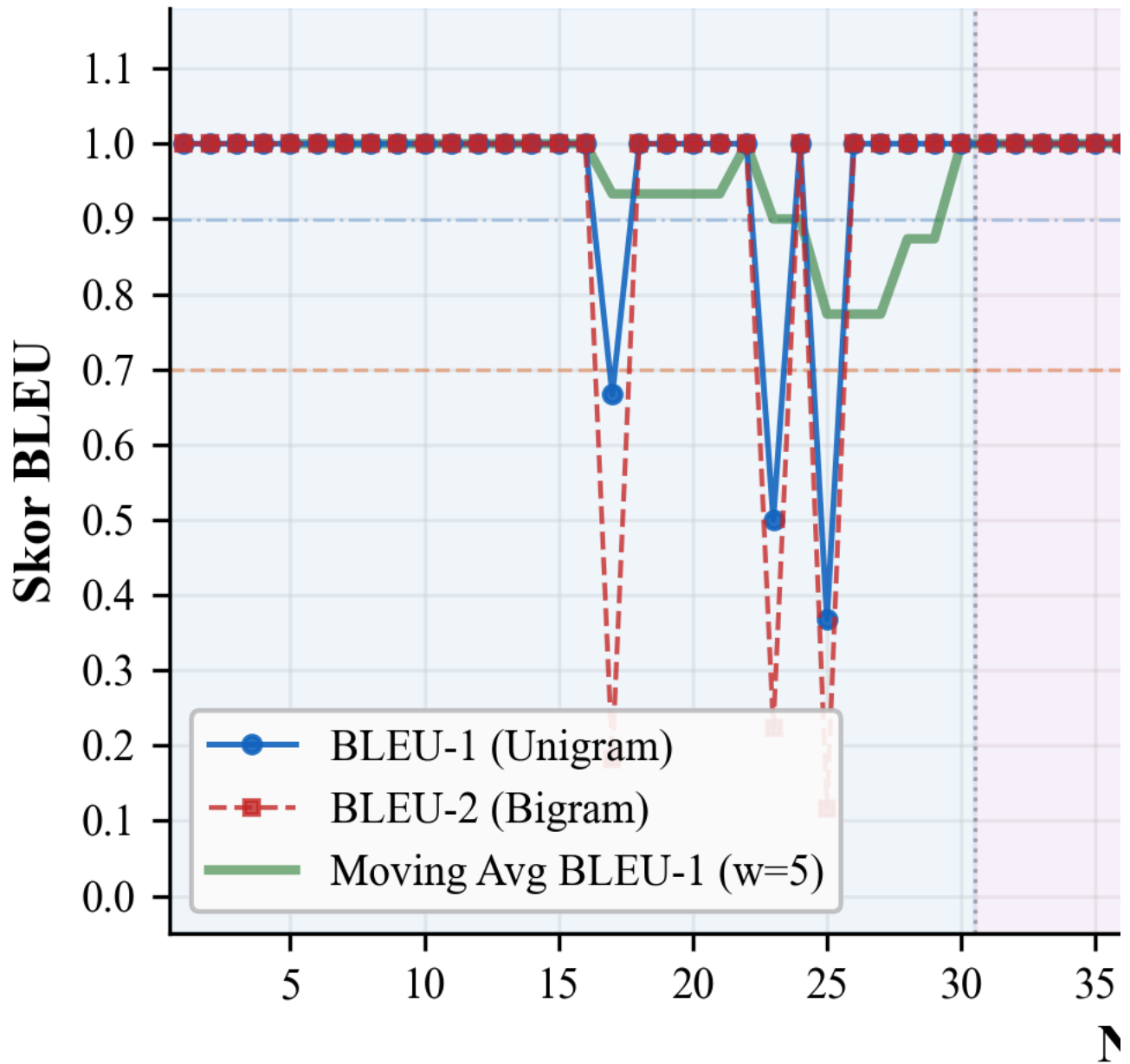
Kategori	N	BLEU-1	BLEU-2	Kualitas
A. Kalimat Terstruktur	30	<b>0.9512</b>	<b>0.9174</b>	Sangat Baik
B. Kalimat Acak	25	<b>0.7294</b>	<b>0.6983</b>	Sangat Baik
C. Kata Tunggal	26	<b>1.0000</b>	<b>0.4740</b>	Sempurna (BLEU-1)
<b>OVERALL AVERAGE</b>	<b>81</b>	<b>0.8984</b>	<b>0.7074</b>	<b>Sangat Baik</b>

### 9.4 Visualisasi Hasil

Berikut adalah grafik performa BLEU Score yang dihasilkan dari pengujian:

# Grafik BLEU Score per (81 Test Case)

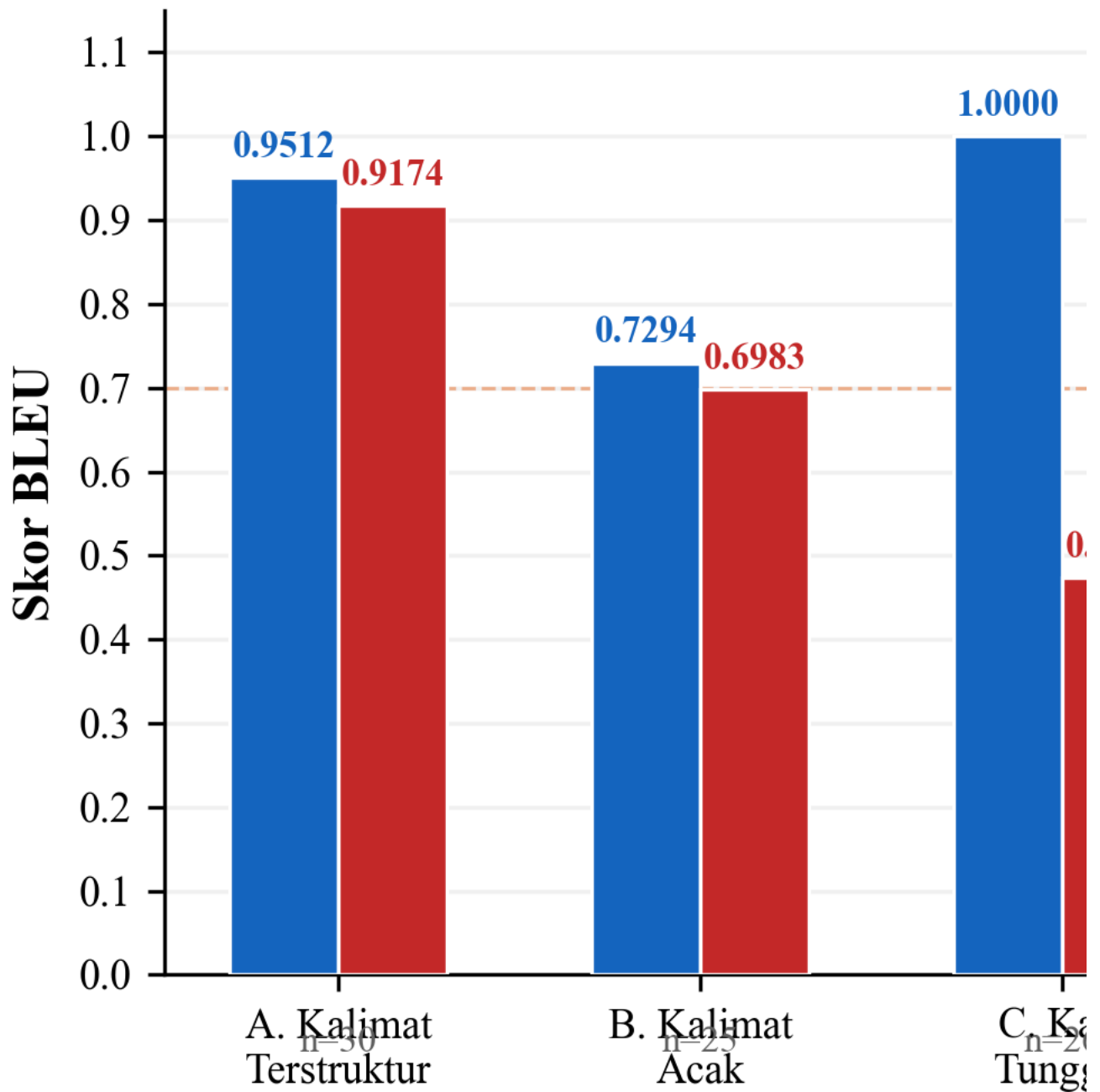
## Kat. A: Terstruktur (n=30)



Gambar 9.1: Grafik skor BLEU-1 dan BLEU-2 untuk setiap test case, menunjukkan konsistensi performa di berbagai kategori.

2. Grafik Bar Chart - Rata-rata per Kategori

## Rata-rata BLEU Score per Kategori —



Gambar 9.2: Perbandingan rata-rata skor BLEU antar kategori pengujian.

**Statistik Tambahan:** - Test case dengan BLEU-1 sempurna ( $\geq 0.99$ ): **70/81 (86.4%)** - Test case dengan BLEU-1  $\geq 0.5$ : **73/81 (90.1%)** - Test case dengan BLEU-1  $< 0.3$ : **5/81 (6.2%)**

### 9.5 Contoh Hasil Per Kategori

#### A. Kalimat Terstruktur (Sampel):

Input Token	Output NLG	BLEU-1	BLEU-2
[Saya, Makan]	Saya makan.	1.0000	1.0000
[Dia, Belajar]	Dia belajar.	1.0000	1.0000
[Maaf, Saya, Bingung]	Maaf, saya bingung.	1.0000	1.0000
[Berapa, Tinggi, Kamu]	Berapa tinggi kamu?	1.0000	1.0000
[Selamat, Pagi]	Selamat pagi!	1.0000	1.0000

#### B. Kalimat Acak - Menguji SPOK Reordering (Sampel):

Input Token (Acak)	Output NLG (Reordered)	BLEU-1	BLEU-2
[Makan, Saya]	Saya makan.	1.0000	1.0000
[Tidur, Kamu]	Kamu tidur.	1.0000	1.0000
[Bingung, Saya, Maaf]	Maaf, saya bingung.	1.0000	1.0000
[Tinggi, Berapa, Dia]	Berapa tinggi dia?	1.0000	1.0000
[Belajar, Halo, Saya]	Halo! Saya belajar.	1.0000	1.0000

### 9.5 Analisis Kegagalan

Terdapat 5 test case dengan BLEU-1 = 0.0 (semua pada kategori Kalimat Acak), yaitu compound word yang ditulis terbalik:

Input (Terbalik)	Output	Referensi	Penyebab
[Kasih, Terima]	(kosong)	Terima kasih!	Compound tidak terdeteksi dalam urutan terbalik
[Pagi, Selamat]	(kosong)	Selamat pagi!	Compound tidak terdeteksi dalam urutan terbalik
[Siang, Selamat]	(kosong)	Selamat siang!	Compound tidak terdeteksi dalam urutan terbalik
[Sore, Selamat]	(kosong)	Selamat sore!	Compound tidak terdeteksi dalam urutan terbalik
[Malam, Selamat]	(kosong)	Selamat malam!	Compound tidak terdeteksi dalam urutan terbalik

**Analisis:** Kegagalan terjadi karena `COMPOUND_MAP` hanya mencocokkan pasangan token dalam urutan yang benar (misalnya Selamat+Pagi), bukan dalam urutan terbalik (Pagi+Selamat). Token individu "Kasih", "Pagi", "Siang", "Sore", dan "Malam" tidak terdapat dalam daftar `LABELS` resmi sehingga di-filter oleh `canonicalize_tokens()`.

### 9.6 Interpretasi dan Kesimpulan

Rentang BLEU	Kategori
0.0 - 0.1	Sangat rendah
0.1 - 0.3	Rendah
0.3 - 0.5	Cukup baik
0.5 - 0.7	Baik
0.7 - 1.0	Sangat baik

**Kesimpulan:** NLG Mode Kalimat memperoleh skor **Overall BLEU-1 = 0.8984** dan **BLEU-2 = 0.7074**, yang masuk dalam kategori **Sangat Baik**. Sistem berhasil menghasilkan kalimat natural yang sangat mirip dengan referensi pada 86.4% test case. Kemampuan Grammar Scoring (SPOK Reordering) terbukti efektif dengan BLEU-1 = 0.7294 pada input token acak. Kelemahan utama terletak pada penanganan compound word dalam urutan terbalik yang dapat diperbaiki dengan menambahkan reverse mapping pada `COMPOUND_MAP`.