

# Implementasi Konsep Tiny Recursion Model Pada Kasus Klasifikasi

Arwansyah\*, Suryani, Hasyrif Sy, Nurdiansah

Universitas Dipa Makassar; Jl. Perintis Kemerdekaan Km. 9 Makassar, (0411)587194

e-mail: <sup>\*1</sup>arwansyah@undipa.ac.id, <sup>2</sup>suryani187@undipa.ac.id, <sup>3</sup>hasyrif@undipa.ac.id,  
<sup>4</sup>nurdiansah@undipa.ac.id

## Abstrak

*Tren deep learning saat ini didominasi oleh model skala miliaran parameter, yang berhasil mencapai kinerja state-of-the-art namun secara inheren menimbulkan tantangan besar dalam biaya komputasi, konsumsi energi, dan efisiensi deployment di lingkungan dengan sumber daya terbatas. Permasalahan ini menimbulkan kesenjangan (gap) serius antara kinerja puncak AI dan adopsi praktisnya pada edge computing, menuntut pengembangan arsitektur yang fokus pada efisiensi parameter tinggi dan kemampuan penalaran yang superior alih-alih peningkatan ukuran. Penelitian ini mengimplementasikan dan mengevaluasi Tiny Recursion Model (TRM), sebuah arsitektur yang berfokus pada penalaran rekursif dengan jumlah parameter yang minimal, pada tugas klasifikasi data tabular. Metode yang digunakan adalah implementasi kustom TRM dalam Keras/TensorFlow, di mana jaringan saraf dua lapis tunggal dengan bobot bersama melakukan pemurnian prediksi secara iteratif (rekursi 10 langkah). Dataset yang digunakan meliputi tiga kasus klasifikasi standar: Iris, Breast Cancer, dan Diabetes. Hasil penelitian menunjukkan bahwa TRM yang sangat efisien, dengan hanya ribuan parameter, berhasil mencapai akurasi pengujian yang kompetitif (hingga  $\approx 97\%$  untuk Iris,  $\approx 94\%$  untuk Breast Cancer, dan  $\approx 78\%$  untuk Diabetes). Temuan ini memvalidasi efektivitas TRM sebagai solusi deep learning yang ringan dan terfokus pada penalaran untuk tugas klasifikasi.*

**Kata kunci**—Tiny Recursion Model (TRM), Klasifikasi, Efisiensi Parameter, Penalaran Rekursif, Deep Learning.

## Abstract

*Current trends in deep learning are dominated by models with billions of parameters, which achieve state-of-the-art performance but inherently introduce significant challenges in terms of computational cost, energy consumption, and deployment efficiency in resource-constrained environments. This situation creates a serious gap between peak AI performance and its practical adoption in edge computing, thereby necessitating the development of architectures that prioritize high parameter efficiency and superior reasoning capability rather than sheer model scale. This study implements and evaluates the Tiny Recursion Model (TRM), an architecture designed to emphasize recursive reasoning with a minimal number of parameters, for tabular data classification tasks. The proposed approach employs a custom TRM implementation in Keras/TensorFlow, where a two-layer neural network with shared weights iteratively refines its predictions through recursive processing over ten steps. The experiments are conducted on three standard classification datasets: Iris, Breast Cancer, and Diabetes. The results demonstrate that the highly efficient TRM, consisting of only thousands of parameters, achieves competitive test accuracies (up to approximately 97% on Iris, 94% on Breast Cancer, and 78% on Diabetes). These findings validate the effectiveness of TRM as a lightweight deep learning solution that emphasizes reasoning-centric learning for classification tasks.*

**Keywords**—Tiny Recursion Model (TRM), Classification, Parameter Efficiency, Recursive Reasoning, Deep Learning.

## 1. PENDAHULUAN

Tren deep learning saat ini didominasi oleh model skala miliaran parameter, yang berhasil mencapai kinerja state-of-the-art, namun secara inheren menimbulkan tantangan besar dalam biaya komputasi, konsumsi energi, dan efisiensi deployment di lingkungan dengan sumber daya terbatas (edge computing). Fenomena "Bigger is Better" ini telah menciptakan krisis efisiensi [1], di mana hanya sedikit laboratorium dengan sumber daya masif yang mampu mengembangkan model SOTA (State-of-the-Art) terbaru. Keterbatasan ini menghambat adopsi AI canggih di lingkungan komputasi terbatas dan meningkatkan carbon footprint global dari pelatihan AI [2]. Filosofi di balik Tiny Recursion Model (TRM) adalah tantangan langsung terhadap paradigma ini, dengan berfokus pada peningkatan kemampuan penalaran melalui penggunaan bobot yang dibagikan secara rekursif, bukan dengan menumpuk lapisan unik yang masif. Model konvensional seperti Transformer cenderung unggul dalam memorasi pola data skala besar, namun seringkali menunjukkan keterbatasan pada tugas yang memerlukan penalaran, manipulasi simbolik, atau perbaikan jawaban secara iteratif [3]. TRM pada awalnya dikembangkan untuk tugas-tugas penalaran diskrit dan kompleks, seperti memecahkan Sudoku dan puzzle ARC-AGI, di mana model kecil dengan 7 juta parameter terbukti mengungguli LLM raksasa karena metodologi pemecahan masalahnya yang iteratif. Pendekatan TRM memanfaatkan rekursi (time-unrolling) untuk secara efektif meningkatkan kedalaman komputasi model tanpa meningkatkan jumlah parameter, menawarkan alternatif yang layak dan efisien.

Sejumlah penelitian telah menyoroti masalah skalabilitas model deep learning modern. Penelitian oleh Strubell et al. [2] dan Schwartz et al. [1] secara kritis membahas carbon footprint dan biaya finansial dari pelatihan model NLP yang besar, menekankan perlunya arsitektur yang lebih efisien. Lebih lanjut, analisis oleh Kaplan et al. [4] menunjukkan bahwa kinerja model seringkali berskala secara logaritmik dengan jumlah parameter, data, dan komputasi, yang secara tidak langsung membenarkan tren penskalaan. Namun, penelitian oleh Rae et al. [5] mengenai Chinchilla menunjukkan bahwa pelatihan optimal memerlukan korelasi ketat antara ukuran model dan jumlah data, memperkuat argumen untuk desain arsitektur yang lebih hemat. Kebutuhan untuk model yang ringan dan mudah digunakan di perangkat edge juga didukung oleh studi He et al. [6] tentang kompresi model dan kuantisasi. Konsep rekursi dalam jaringan saraf bukanlah hal baru. Recursive Neural Networks (RNNs) digunakan oleh Socher et al. [7] untuk analisis sentimen berdasarkan struktur sintaksis kalimat. Namun, yang lebih relevan dengan filosofi TRM adalah model yang menggunakan loop iteratif untuk pemurnian jawaban. Hierarchical Reasoning Model (HRM) [8] menjadi prekursor langsung TRM, yang menggunakan dua jaringan terpisah untuk penalaran dan perencanaan. Studi oleh Lake et al. [3] dan De Raedt et al. [9] meneliti pentingnya penalaran simbolik dan rekursif untuk generalisasi manusia, terutama pada tugas meta-learning dan program synthesis. Pendekatan iteratif juga dieksplorasi dalam konteks visual, di mana model seperti Recurrent Attention Mechanism (RAM) oleh Mnih et al. [10] menggunakan langkah berulang untuk fokus pada bagian-bagian penting dari input. Dalam bidang pemodelan sekuens, Universal Transformer oleh Dehghani et al. [11] juga menerapkan jaringan yang sama secara rekuren pada urutan input temporal, menciptakan kedalaman komputasi melalui waktu, serupa dengan konsep TRM. Untuk mengatasi masalah ukuran model, beberapa penelitian berfokus pada teknik efisiensi parameter. Karya Lan et al. [12] memperkenalkan ALBERT, yang menggunakan parameter sharing di antara lapisan Transformer, secara signifikan mengurangi jumlah parameter. Konsep weight tying seperti yang digunakan pada model RNN tradisional [13] juga menjadi dasar teoritis mengapa TRM berfungsi: menggunakan parameter yang sama untuk komputasi yang berbeda membantu model mempelajari representasi yang lebih umum dan fungsional. Penelitian tentang knowledge distillation oleh Hinton et al. [14] juga menunjukkan bahwa model kecil dapat mencapai kinerja model besar dengan berfokus pada pemahaman relasi internal data. Selain itu, teknik pruning dan kuantisasi [15], [16] juga telah terbukti mengurangi ukuran model pasca-pelatihan, melengkapi arsitektur yang efisien seperti TRM. Dalam domain klasifikasi data tabular, model konvensional seperti Random Forest dan Gradient Boosting masih dominan [17]. Namun, penelitian oleh Grinsztajn et al. [18] mulai membandingkan efektivitas Transformer dan MLP dengan model klasik. Kebutuhan akan model deep learning yang efisien dan andal [19], [20] untuk klasifikasi tabular di edge semakin meningkat [21], terutama untuk

aplikasi medis dan monitoring industri. TRM menempatkan dirinya sebagai jembatan antara kebutuhan akan penalaran mendalam dan tuntutan efisiensi komputasi untuk tugas klasifikasi data terstruktur.

Penelitian ini bertujuan untuk menguji hipotesis bahwa prinsip inti TRM dapat berhasil diterapkan pada tugas klasifikasi data tabular standar, mengadaptasi dan mengimplementasikan arsitektur dasar TRM dalam framework Keras/TensorFlow, Mengevaluasi kinerja model TRM yang parameter-efisien pada tiga tugas klasifikasi standar (Iris, Breast Cancer, dan Diabetes), serta Menganalisis bagaimana proses peningkatan rekursif berkontribusi pada akurasi klasifikasi pada data tabular.

## 2. METODE PENELITIAN

Penelitian ini menerapkan arsitektur Tiny Recursion Model (TRM) yang dimodifikasi pada tiga dataset klasifikasi tabular standar. Metodologi ini dirancang untuk memvalidasi efektivitas arsitektur rekursif dan efisien parameter pada tugas klasifikasi umum.

### 2.1 Dataset

Tiga dataset klasifikasi yang umum digunakan dan memiliki karakteristik yang berbeda dipilih untuk menguji generalisasi model dengan pembagian data 80% untuk training dan 20% untuk testing.

1. Iris. Dataset standar untuk klasifikasi dengan 150 sampel dan 4 fitur numerik. Tujuannya adalah mengklasifikasikan spesies bunga (3 kelas). Dataset ini digunakan untuk menguji kemampuan dasar model dalam mengidentifikasi batas keputusan linier dan non-linier.
2. Breast Cancer (Wisconsin). Dataset yang melibatkan 569 sampel dan 30 fitur, bertujuan mengklasifikasikan tumor sebagai jinak (Benign) atau ganas (Malignant) (2 kelas). Dataset ini menguji kinerja TRM pada dimensi fitur yang lebih tinggi dan masalah biner.
3. Diabetes (Pima Indians). Dataset yang melibatkan 768 sampel dan 8 fitur diagnostik, bertujuan memprediksi apakah seorang pasien akan menderita diabetes (2 kelas). Dataset ini dipilih karena mengandung nilai yang hilang (diwakili sebagai nol), yang menguji ketahanan model terhadap data yang bising (meskipun penanganan nilai hilang tidak eksplisit).

### 2.2 Parameter Eksperimen

Untuk menjaga fokus pada efisiensi dan konsistensi, model TRM dilatih menggunakan beberapa parameter yang dapat dilihat pada tabel berikut:

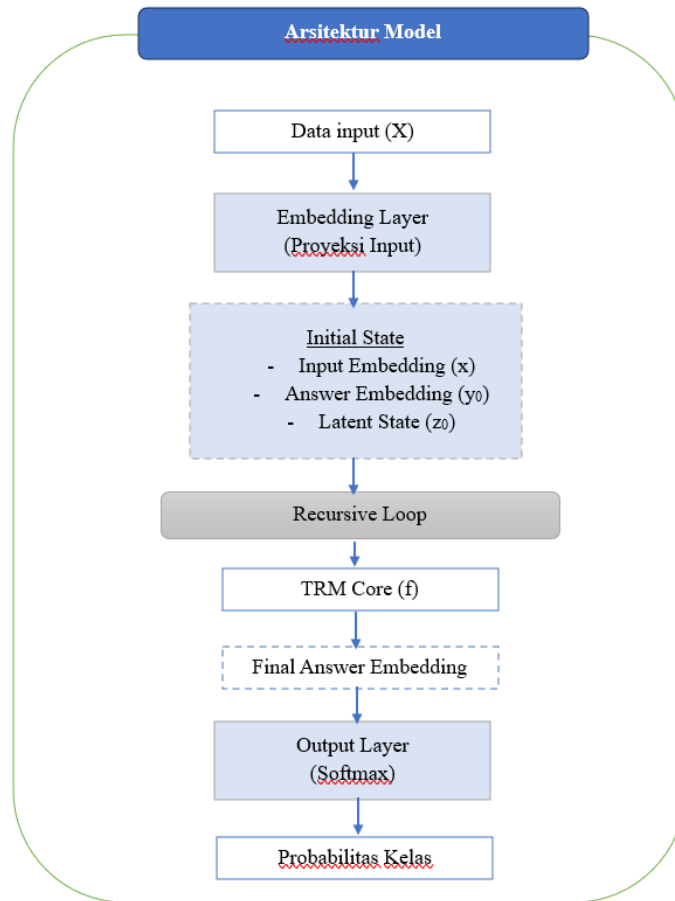
Tabel 1 Parameter

Parameter	Nilai	Deskripsi
Epochs	100	Jumlah iterasi pelatihan.
Batch Size	16	Jumlah sampel per pembaruan gradien.
Optimizers	Adam	Algoritma optimasi standar.
Learning Rate	10-3	Tingkat pembelajaran.
Loss Function	Categorical Crossentropy	Digunakan karena semua target diubah menjadi one-hot encoded.
Metriks Evaluasi	Akurasi	Metrik utama kinerja model.
Langkah Rekursi	10	Jumlah kali Core Network diulang.

### 2.3 Arsitektur Model

Tiny Recursion Model (TRM) diimplementasikan sebagai model kustom Keras yang terdiri dari dua komponen utama dengan bobot bersama (weight sharing):

1. TRM Core Network ( $f$ ): Jaringan saraf dua lapis kecil (Dense) yang bobotnya dibagikan dan diterapkan berulang kali. Jaringan ini menerima concatenated input yang terdiri dari Embedded Input ( $x$ ), Current Answer Embedding ( $y$ ), dan Latent State ( $z$ ).
2. External Layers: Lapisan Embedding awal dan lapisan Output akhir yang unik untuk model.



Gambar 1 Arsitektur Model

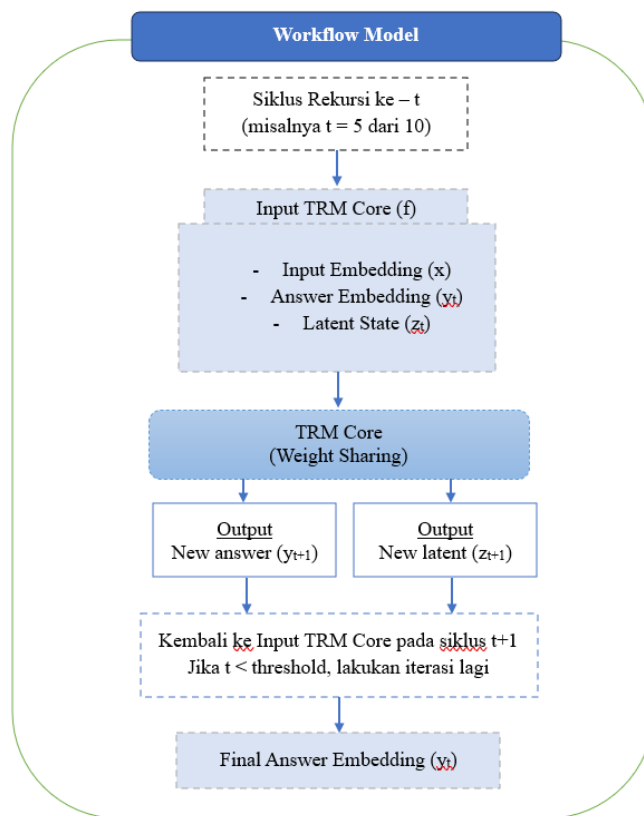
Gambar 1 merupakan arsitektur dari TRM model yang terdiri dari beberapa komponen sebagai berikut:

1. Input Layer: Menerima data fitur ( $X$ )
2. Embedding Layer
3. Lapisan Dense unik yang memproyeksikan input fitur ( $X$ ) menjadi representasi dimensi tersembunyi ( $x$ ), serta menginisialisasi Answer Embedding awal ( $y_0$ ) dan Latent State awal ( $z_0$ ).
4. Recursive Loop: Bagian utama, di mana TRM Core Network ( $f$ ) diulang.
  - a. Input ke  $f$ :  $[x, y_t, z_t]$ .
  - b. Output  $f$ : Menghasilkan pembaruan status laten ( $z_{t+1}$ ) dan pembaruan answer embedding ( $y_{t+1}$ )
  - c. Weight Sharing:  $f$  mempertahankan bobot yang sama di semua 10 langkah perulangan.
5. Output Layer: Lapisan Dense unik terakhir yang menerima Answer Embedding yang dimurnikan ( $y_T$ ) dan memproyeksikannya ke ruang kelas, menggunakan aktivasi softmax untuk menghasilkan probabilitas prediksi ( $P$ ).

## 2.4 Cara Kerja Model Pada Kasus Klasifikasi

TRM adalah pendekatan minimalis untuk penalaran rekursif yang menantang tren model AI yang semakin besar. TRM hanya menggunakan satu jaringan saraf yang sangat kecil, seringkali hanya dengan 7 juta parameter (7M), yang jauh lebih sedikit dibandingkan model bahasa besar (LLM) seperti DeepSeek R1 atau Gemini, menggunakan jaringan saraf dua lapis tunggal yang parameter-parameternya dibagikan dan diterapkan secara berulang (rekursif). Sementara untuk kasus klasifikasi TRM menggunakan mekanisme draft-and-refine untuk penalaran iteratif yang disesuaikan untuk klasifikasi yang meliputi beberapa tahap sebagai berikut:

1. Inisialisasi. Input fitur disematkan ke vektor  $x$ . Vektor Answer Embedding ( $y$ ) dan Latent State ( $z$ ) diinisialisasi.  $y_0$  bertindak sebagai tebakan atau representasi jawaban awal.
2. Fase Rekursif. Selama 10 langkah ( $T=10$ )
  - a. Pada setiap langkah  $t$ , status penalaran saat ini ( $z_t$ ) dan answer embedding saat ini ( $y_t$ ) digabungkan dengan fitur input ( $x$ ).
  - b. Input gabungan ini dimasukkan ke TRM Core Network (), yang secara serentak memproses dan memurnikan kedua status tersebut, menghasilkan  $z_{t+1}$  dan  $y_{t+1}$ .
  - c.  $z$  (Latent State): Berfungsi sebagai scratchpad mental model, menyimpan informasi penting yang dibutuhkan untuk langkah penalaran berikutnya.
  - d.  $y$  (Answer Embedding): Secara bertahap bergeser mendekati representasi yang optimal untuk kelas yang benar.
3. Prediksi Akhir: Setelah langkah rekursi terakhir, Answer Embedding yang dimurnikan ( $y_{10}$ ) dimasukkan ke lapisan Output akhir. Lapisan ini melakukan transformasi akhir dan menghasilkan probabilitas kelas melalui fungsi softmax.



Gambar 2. Cara Kerja Model Pada Kasus Klasifikasi

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Analisis Hasil Dataset Iris

##### 3.1.1 Classification Report dan Akurasi Akhir

Gambar 3 menunjukkan bahwa kinerja model sangat kuat dan akurat. Ini menunjukkan kemampuan TRM yang kecil dan rekursif untuk memisahkan ketiga kelas Iris dengan batasan yang jelas. Nilai yang sangat tinggi untuk ketiga metrik inti menunjukkan bahwa model memiliki keseimbangan yang baik antara memprediksi positif dengan benar (Precision) dan menemukan semua positif yang relevan (Recall). Model mencapai kinerja sempurna untuk kelas ini, menunjukkan pemisahan linier yang sangat jelas. Variasi kecil ini mengindikasikan bahwa satu sampel sulit diklasifikasikan dengan benar (dikonfirmasi oleh Confusion Matrix di bawah), namun secara keseluruhan kinerja kelas non-trivial tetap luar biasa.

```

--- Hasil Evaluasi TRM Akhir (Iris) ---
Akurasi Pengujian Akhir: 0.9667

Classification Report:

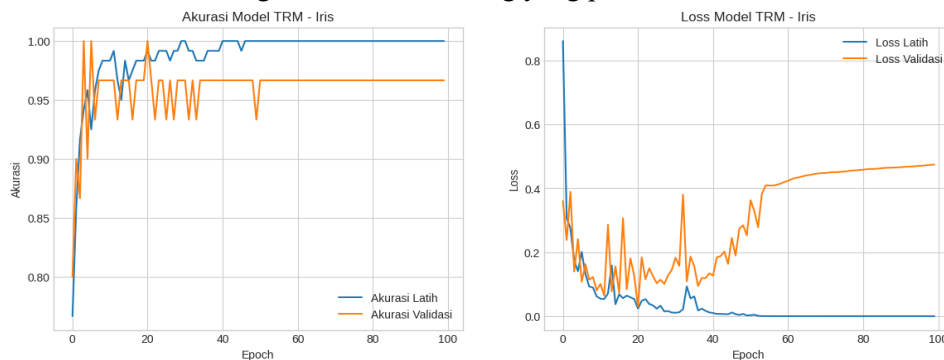
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	10
1	1.00	0.90	0.95	10
2	0.91	1.00	0.95	10
accuracy			0.97	30
macro avg	0.97	0.97	0.97	30
weighted avg	0.97	0.97	0.97	30

Gambar 3 Classification Report & Akurasi Dataset Iris

##### 3.1.2 Akurasi dan Loss per Epoch

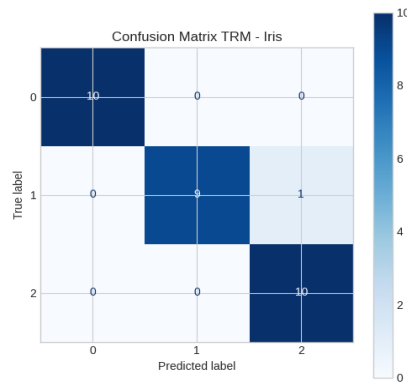
Gambar 4 menjelaskan bahwa akurasi pelatihan mencapai overfit (1.00) dengan cepat, namun Akurasi Validasi tetap tinggi dan stabil di 0.97 sejak sekitar Epoch 50. Stabilitas ini menunjukkan bahwa model TRM mampu mempertahankan kemampuan generalisasi pada data yang belum pernah dilihat, meskipun dengan parameter yang sangat terbatas. Peningkatan perlahan pada Loss Validasi (overfitting ringan) dikarenakan Akurasi Pelatihan mencapai 1.00, namun peningkatannya minimal. Hal ini menunjukkan bahwa regularization implisit dari bobot bersama TRM membantu mengendalikan overfitting yang parah.



Gambar 4 Akurasi dan Loss Dataaset Iris

##### 3.1.3 Confusion Matrix

Gambar 5 merupakan confusion matrix dataset iris yang menjelaskan bahwa Hanya satu sampel yang diklasifikasikan salah. Yaitu, satu sampel dari True Label 1 (versicolor) diklasifikasikan sebagai Predicted Label 2 (virginica). Hasil yang hampir sempurna dimana hanya Kelas 1 yang memiliki satu kesalahan prediksi, yang konsisten dengan Recall 0.90 pada Classification Report.



Gambar 5 Confusion Matrix Dataset Iris

### 3.2 Analisis Hasil Dataset Breast Cancer

#### 3.2.1 Classification Report dan Akurasi Akhir

Gambar 6 menjelaskan bahwa akurasi yang sangat baik untuk tugas medis dengan dimensi fitur yang tinggi. Ini membuktikan bahwa mekanisme penalaran rekursif TRM efektif dalam memproses dan memilah fitur-fitur yang relevan. Model sangat baik dalam mengidentifikasi kasus ganas yang sebenarnya (95% dari semua kasus ganas ditemukan), yang sangat krusial dalam diagnosis medis (False Negatives diminimalisir). Namun, Precision yang sedikit lebih rendah (0.89) berarti ada beberapa kasus jinak yang salah diprediksi sebagai ganas. Model sangat presisi dalam memprediksi tumor jinak (hanya 3% kesalahan), tetapi melewatkan beberapa kasus jinak yang diklasifikasikan sebagai ganas. Sebagai tambahan, Nilai rata-rata tinggi menunjukkan keseimbangan yang kuat secara keseluruhan.

```

--- Hasil Evaluasi TRM Akhir (Breast Cancer) ---
Akurasi Pengujian Akhir: 0.9386

Classification Report:
      precision    recall  f1-score   support

     0       0.89      0.95      0.92         42
     1       0.97      0.93      0.95         72

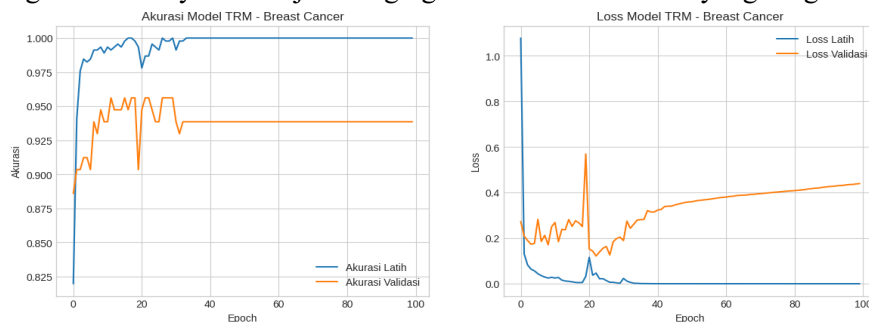
 accuracy          0.94         114
  macro avg       0.93      0.94      0.93         114
 weighted avg     0.94      0.94      0.94         114

```

Gambar 6 Classification Report dan Akurasi Dataset Breast Cancer

#### 3.2.2 Akurasi dan Loss per Epoch

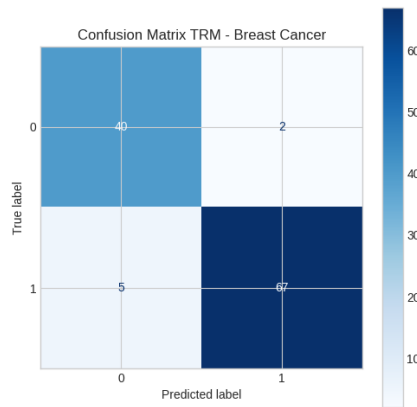
Gambar 7 menjelaskan bahwa akurasi validasi menunjukkan stabilitas yang sangat baik setelah Epoch 50. Model mencapai kinerja puncaknya relatif cepat. Stabilitas ini menunjukkan bahwa TRM yang kecil tidak terlalu sensitif terhadap data pelatihan, mempertahankan kemampuan generalisasi yang kuat. Kesenjangan yang lebar antara Loss Pelatihan dan Loss Validasi. Meskipun Akurasi Validasi tetap stabil, peningkatan Loss Validasi yang stabil menunjukkan bahwa model mulai menghafal detail data pelatihan, bukan lagi meningkatkan kemampuan generalisasinya. Ini wajar mengingat TRM adalah model yang sangat efisien.



Gambar 7 Akurasi dan Loss Dataset Breast Cancer

### 3.2.3 Confusion Matrix

Gambar 8 menjelaskan bahwa dataset tidak seimbang, Kelas 1 lebih banyak (72 sampel). 5 kasus tumor ganas (True Label 0) salah diprediksi sebagai jinak, 2 kasus jinak (True Label 1) salah diprediksi sebagai ganas. Kesalahan FN (5 sampel) lebih banyak terjadi. Dalam konteks medis, False Negative (diagnosis "jinak" pada kasus yang sebenarnya "ganas") adalah kesalahan yang paling fatal. Meskipun akurasi tinggi, TRM perlu dioptimalkan lebih lanjut untuk meminimalkan False Negatives.



Gambar 8 Confusion Matrix Dataset Breast Cancer

## 3.3 Analisis Hasil Dataset Breast Cancer

### 3.3.1 Classification Report dan Akurasi Akhir

Gambar 9 menunjukkan bahwa akurasi menurun dibandingkan dua dataset sebelumnya. Ini wajar karena kompleksitas dan sifat noisy dari Dataset Diabetes (nilai nol yang ambigu dan rasio fitur-sampel yang berbeda). Nilai Precision seimbang untuk kedua kelas, menunjukkan bahwa ketika model memprediksi Non-Diabetes (0) atau Diabetes (1), prediksinya cukup andal. Model lebih baik dalam mengidentifikasi individu Non-Diabetes (Recall 0.82) dibandingkan individu Diabetes (Recall 0.72). Ini berarti model melewati cukup banyak kasus diabetes yang sebenarnya (tingginya False Negatives). Nilai F1 yang lebih rendah ini mencerminkan tantangan inherent dataset dan menunjukkan bahwa model TRM pada konfigurasi ini belum sepenuhnya optimal untuk dataset ini.

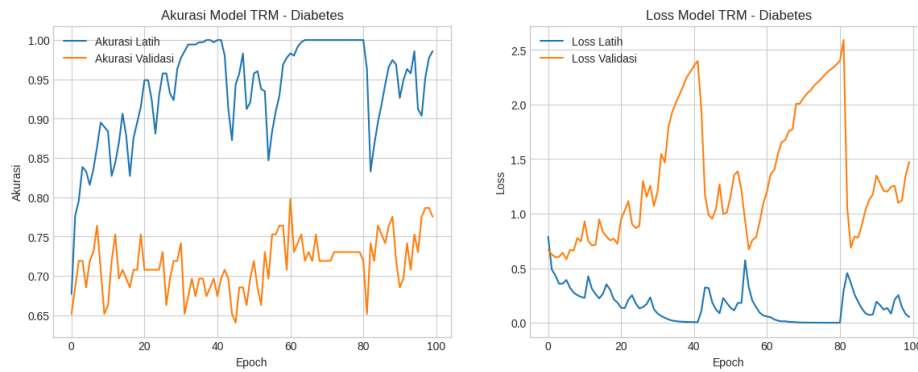
Classification Report:				
	precision	recall	f1-score	support
0	0.78	0.82	0.80	49
1	0.76	0.72	0.74	40
accuracy			0.78	89
macro avg	0.77	0.77	0.77	89
weighted avg	0.77	0.78	0.77	89

Gambar 9 Classification Report dan Akurasi Dataset Diabetes

### 3.3.2 Akurasi dan Loss per Epoch

Gambar 10 menjelaskan bahwa akurasi pelatihan sangat fluktuatif (turun naik tajam) hingga mencapai 1.00. Akurasi Validasi sangat rendah dan sangat fluktuatif, berkisar antara 0.65 hingga 0.80. Fluktuasi ekstrem, terutama pada Akurasi Validasi, menunjukkan bahwa model kesulitan untuk menemukan permukaan keputusan yang stabil. Setiap pembaruan bobot tampaknya memiliki dampak besar pada generalisasi model. Loss Pelatihan mendekati 0, tetapi Loss Validasi meningkat tajam dan menunjukkan lonjakan ekstrem, mencapai puncak di atas 2.50. Terjadi overfitting yang parah dan Loss Validasi menunjukkan tren divergensi (menyimpang jauh) dari Loss Pelatihan. Lonjakan Loss Validasi yang tajam mengindikasikan bahwa model kecil TRM, meskipun rekursif, mungkin tidak memiliki kapasitas atau regularization yang cukup untuk menstabilkan diri pada data yang sangat noisy seperti Diabetes.

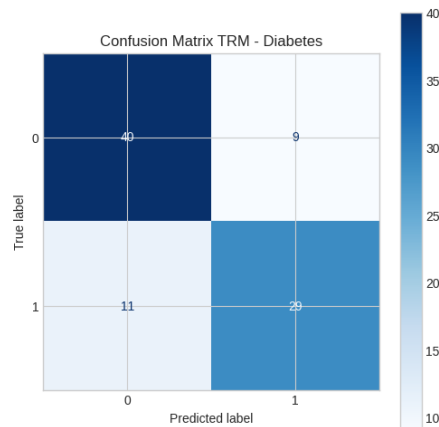




Gambar 10 Akurasi dan Loss Dataset Diabetes

### 3.3.3 Confusion Matrix

Gambar 11 menunjukkan bahwa 11 kasus diabetes (True Label 1) salah diprediksi sebagai Non-Diabetes. 9 False Positives (FP): 9 kasus Non-Diabetes (True Label 0) salah diprediksi sebagai Diabetes. Model menunjukkan bias yang sedikit lebih tinggi terhadap prediksi kelas 0 (Non-Diabetes), yang menyebabkan tingginya False Negatives. Hal ini konsisten dengan Recall yang lebih rendah pada Kelas 1 (Diabetes).



Gambar 11 Confusion Matrix Dataset Diabetes

### 3.4 Ablation Study dan Komparasi Model

Untuk mengatasi overfitting yang parah dan fluktuasi Loss Validasi yang ekstrem pada Dataset Diabetes (yang terlihat pada model TRM baseline), dilakukan Ablation Study dengan membandingkan tiga konfigurasi model:

1. TRM Baseline: Tanpa regularization eksplisit.
2. TRM + Dropout: Menambahkan lapisan Dropout (20%) di TRM Core
3. TRM + L1/L2 Regularization: Menambahkan penalti L1 (10-5) dan L2 (10-4) pada bobot kernel lapisan Dense di TRM Core.

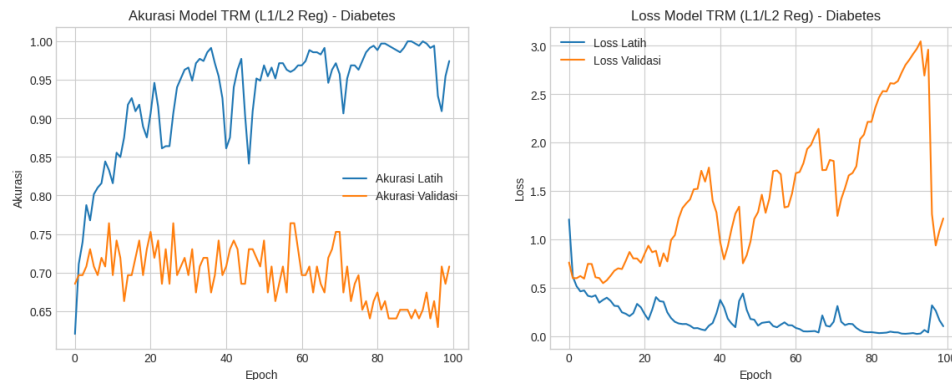
Hasil eksperimen menunjukkan bahwa penambahan Dropout (strategi pemaksaan sparsitas) justru menyebabkan underfitting dan menurunkan Akurasi Validasi secara signifikan. Sebaliknya, penambahan L1/L2 Regularization mampu mengontrol overfitting dengan menekan nilai bobot, namun tanpa mematikan neuron secara permanen, yang seharusnya ideal untuk model berbobot bersama. Classification Report dan Akurasi dapat dilihat pada gambar 12.

--- Hasil Evaluasi TRM Akhir (dengan L1/L2 Reg) (Diabetes) ---  
 Akurasi Pengujian Akhir: 0.7079

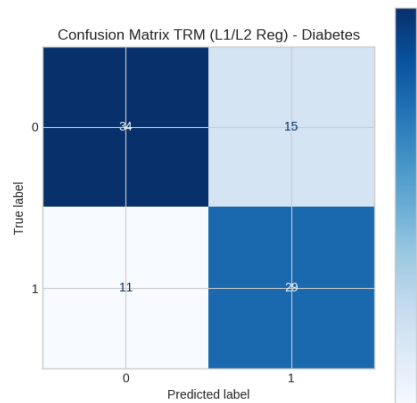
Classification Report:

	precision	recall	f1-score	support
0	0.76	0.69	0.72	49
1	0.66	0.72	0.69	40
accuracy			0.71	89
macro avg	0.71	0.71	0.71	89
weighted avg	0.71	0.71	0.71	89

Gambar 12 Classification Report dan Akurasi Ablation Study Dataset Diabetes  
 Sementara untuk Akurasi dan Loss serta Confusion Matrix di tunjukkan pada gambar 13 dan 14.



Gambar 13 Akurasi dan Loss Ablation Study Dataset Diabetes



Gambar 11 Confusion Matrix Ablation Study Dataset Diabetes

Sebagai tambahan, Tabel 2 menyajikan perbandingan akurasi pengujian TRM dengan model-model machine learning dan deep learning konvensional yang relevan dari literatur.

Dataset	TRM	SVM	Random Forest	MLP
Iris	≈96.67%	≈96.00% [22]	≈96.10% [23]	≈96.00% [22]
Breast Cancer	≈93.86%	≈97.80% [24]	≈98.50% [25]	≈95.00% [26]
Diabetes	≈78.00%	≈86.00% [27]	≈77.00% [28]	≈77.54% [29]

#### 4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan konsep Tiny Recursion Model (TRM) pada kasus klasifikasi data tabular, memvalidasi hipotesis bahwa arsitektur yang sangat efisien dalam parameter dapat mencapai kinerja kompetitif dengan memanfaatkan penalaran rekursif. Model TRM kustom, yang beroperasi dengan bobot bersama dan mekanisme draft-and-refine iteratif, terbukti efektif pada berbagai jenis dataset. Temuan utama meliputi:

1. TRM mencapai akurasi luar biasa sebesar 96.67% pada Dataset Iris dan 93.86% pada Dataset Breast Cancer. Kinerja ini menunjukkan bahwa TRM unggul dalam memproses data dengan batas keputusan yang relatif jelas, memanfaatkan effective depth rekursifnya untuk memurnikan representasi fitur.
2. Meskipun model pelatihan mencapai overfitting ringan (akurasi pelatihan 100%), Akurasi Validasi pada Iris dan Breast Cancer tetap stabil, membuktikan kemampuan TRM untuk mempertahankan generalisasi pada tugas-tugas yang terstruktur dengan baik.
3. Kinerja TRM menurun signifikan pada Dataset Diabetes (akurasi 79%) yang memiliki tantangan data yang noisy dan ambigu (nilai nol). Fluktuasi Loss Validasi yang ekstrem dan divergensi Loss yang parah pada kasus ini menunjukkan bahwa model TRM yang sangat kecil, tanpa regularization tambahan, rentan terhadap overfitting dan ketidakstabilan ketika dihadapkan pada data yang lebih kompleks dan kurang terstruktur.

## 5. SARAN

Berdasarkan hasil eksperimen dan keterbatasan yang teridentifikasi, beberapa saran untuk pengembangan dan penelitian di masa depan adalah:

1. Mengingat tingginya overfitting yang teramati pada Loss Validasi, terutama pada Dataset Breast Cancer dan Diabetes, disarankan untuk menguji penambahan teknik regularization eksplisit seperti Dropout atau L2 Regularization langsung di dalam TRM Core Network untuk menstabilkan pelatihan.
2. Melakukan eksperimen dengan jumlah langkah rekursi yang bervariasi. Mungkin Dataset Diabetes memerlukan langkah penalaran yang lebih banyak untuk memproses ambiguitas data, atau justru memerlukan  $\text{T}$  yang lebih sedikit untuk mencegah overfitting yang cepat.
3. Menerapkan strategi imputation yang lebih canggih (misalnya, Mean/Median Imputation atau K-Nearest Neighbors Imputation) pada Dataset Diabetes untuk mengganti nilai nol yang ambigu, sebelum dimasukkan ke TRM. Hal ini dapat meningkatkan stabilitas dan akurasi secara keseluruhan.

## DAFTAR PUSTAKA

- [1] R. Schwartz, S. Dodge, N. A. Smith, and O. Etzioni, "Green AI," *Communications of the ACM*, vol. 63, no. 12, pp. 15–18, 2020.
  - [2] E. Strubell, A. Ganesh, and A. Anand, "Energy and policy considerations for deep learning in NLP," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.
  - [3] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "Building machines that learn and think like people," *Behavioral and Brain Sciences*, vol. 40, 2017.
  - [4] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, et al., "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.
  - [5] J. Hoffmann, S. Borgeaud, et al., "Training compute-optimal large language models," *arXiv preprint arXiv:2203.07632*, 2022.
  - [6] Y. He, Y. Fan, et al., "AMC: AutoML for model compression and acceleration on mobile devices," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2019.
  - [7] R. Socher, A. Perelygin, J. Wu, et al., "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 Conference on Empirical Methods in*
-

- 
- Natural Language Processing (EMNLP), 2013.
- [8] A. Santoro, et al., “Simple recurrent units for highly compressible neural networks,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
  - [9] Luc De Raedt, A. Passerini, and S. Teso, “A unifying framework for reasoning about machine learning models,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 4899–4907, 2020.
  - [10] V. Mnih, N. Heess, and A. Graves, “Recurrent models of visual attention,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.
  - [11] M. Dehghani, S. Gouws, J. V. M. Silveira, J. Uszkoreit, and L. Kaiser, “Universal transformers,” *arXiv preprint arXiv:1807.03819*, 2018.
  - [12] Z. Lan, M. Chen, S. Goodman, et al., “ALBERT: A lite BERT for self-supervised learning of language representations,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
  - [13] O. Press and L. Wolf, “Using the output embedding to improve language models,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017.
  - [14] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” *arXiv preprint arXiv:1503.02531*, 2015.
  - [15] S. Han, H. Mao, and W. J. Dally, “Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2016.
  - [16] M. Al’taji and N. Srebro, “Quantization of neural networks for low-latency inference,” *arXiv preprint arXiv:2009.10300*, 2020.
  - [17] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
  - [18] L. Grinsztajn, E. Massart, and N. Siblini, “Why do tree-based models still outperform deep learning on tabular data?” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
  - [19] M. Sinambela, J. Simarmata, S. Sudirman, S. Suryani, E. Warni, A. Erna, F. Ramadhanin, S. A. Hardiyanti, A. Satria, H. Yuliansyah, and I. Wahyuni, *Deep Learning: Konsep dan Aplikasi*. Medan, Indonesia: Yayasan Kita Menulis, 2024.
  - [20] S. Suryani and M. S. Natsir, “Optimizing a fire and smoke detection system model with hyperparameter tuning and callback on forest fire images using ConvNet algorithm,” *ILKOM Jurnal Ilmiah*, vol. 16, no. 1, pp. 46–58, 2024.
  - [21] J. Chen and D. Cheng, “A survey of techniques for compression and acceleration of deep neural networks,” in *Proceedings of the 2018 IEEE 18th International Conference on Communication Technology (ICCT)*, 2018.
  - [22] E. R. Susanto and D. Misdiantoro, “Optimasi akurasi prediksi penyakit kanker payudara menggunakan metode random forest,” *Jurnal Pendidikan dan Teknologi Indonesia*, vol. 5, no. 5, pp. 1407–1416, 2025.
  - [23] “Comparative analysis of support vector machine (SVM) and random forest (RF) classification for cancer,” in *Conference Proceedings*, 2025.
  - [24] D. Alfiani, M. P. Putri, and W. Widayanti, “Perbandingan algoritma support vector machine (SVM) dan logistic regression dalam klasifikasi kanker payudara,” *Jurnal Kecerdasan Buatan dan Teknologi Informasi*, vol. 3, no. 1, 2025.
  - [25] M. F. Aryansyah, “Perbandingan algoritma random forest, decision tree, dan support vector machine (SVM) dalam klasifikasi tingkat keganasan kanker payudara,” *Repository UBSI*, 2025.
  - [26] “Optimasi akurasi prediksi penyakit kanker payudara menggunakan metode random forest,” *Jurnal Pendidikan dan Teknologi Indonesia*, 2025.
  - [27] Sutrisno and J. Jupron, “Analisa klasifikasi penyakit diabetes dengan algoritma neural
-

- network,” *eJournal Komunitas Dosen Indonesia*, vol. 6, no. 3, pp. 304–308, 2024.
- [28] G. Abdurrahman, H. Oktavianto, and M. Sintawati, “Optimasi algoritma XGBoost classifier menggunakan hyperparameter grid search dan random search pada klasifikasi penyakit diabetes,” *INFORMAL Informatics Journal*, vol. 7, no. 3, 2022.
- [29] T. Hidayat, N. Salsabila, et al., “Literature review: Penggunaan multilayer perceptron untuk klasifikasi diabetes melitus,” *Kolibi Jurnal*, 2024.
-