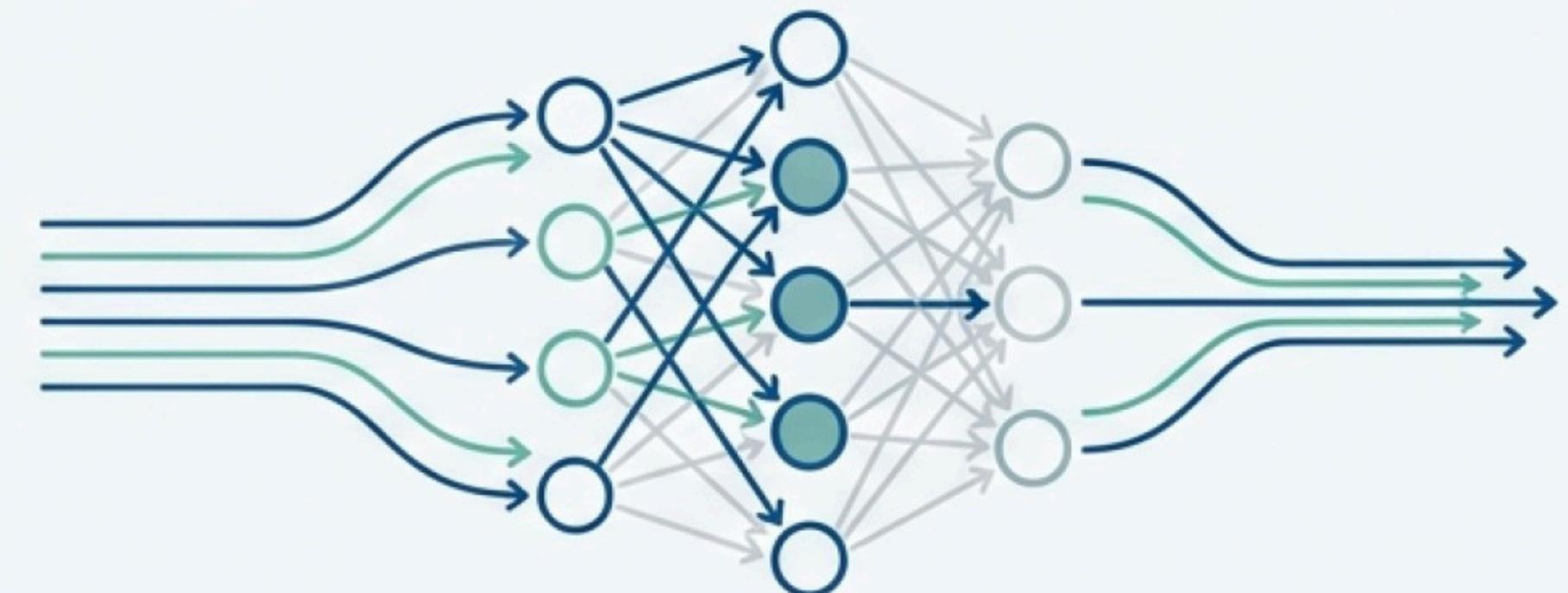


Gerçek Zamanlı Balık Sınıflama: Otomasyon ile Verimliliği Artıran Akıllı Sistem

1D-CNN Mimarisi Kullanılarak Düşük
Gecikmeli Tartım ve Sıralama Çözümü
Geliştirme Raporu



Mevcut Sıralama Sistemi Operasyonel Verimsizliklere Yol Açıyor

Mevcut sistem, balıkların hangi kapıya gönderileceğini manuel olarak tanımlanan statik ağırlık sınırlarına göre belirler. Bu yaklaşımın sorunları:

- **Dinamik Koşulları Göz Ardı Etme**

Balık popülasyonu, yemleme durumu ve eşzamanlı çalışan tartım sistemleri (System A / System B) gibi değişkenleri hesaba katmaz.

- **Operasyonel Sonuçlar**

- **Dengesiz Dağılım**

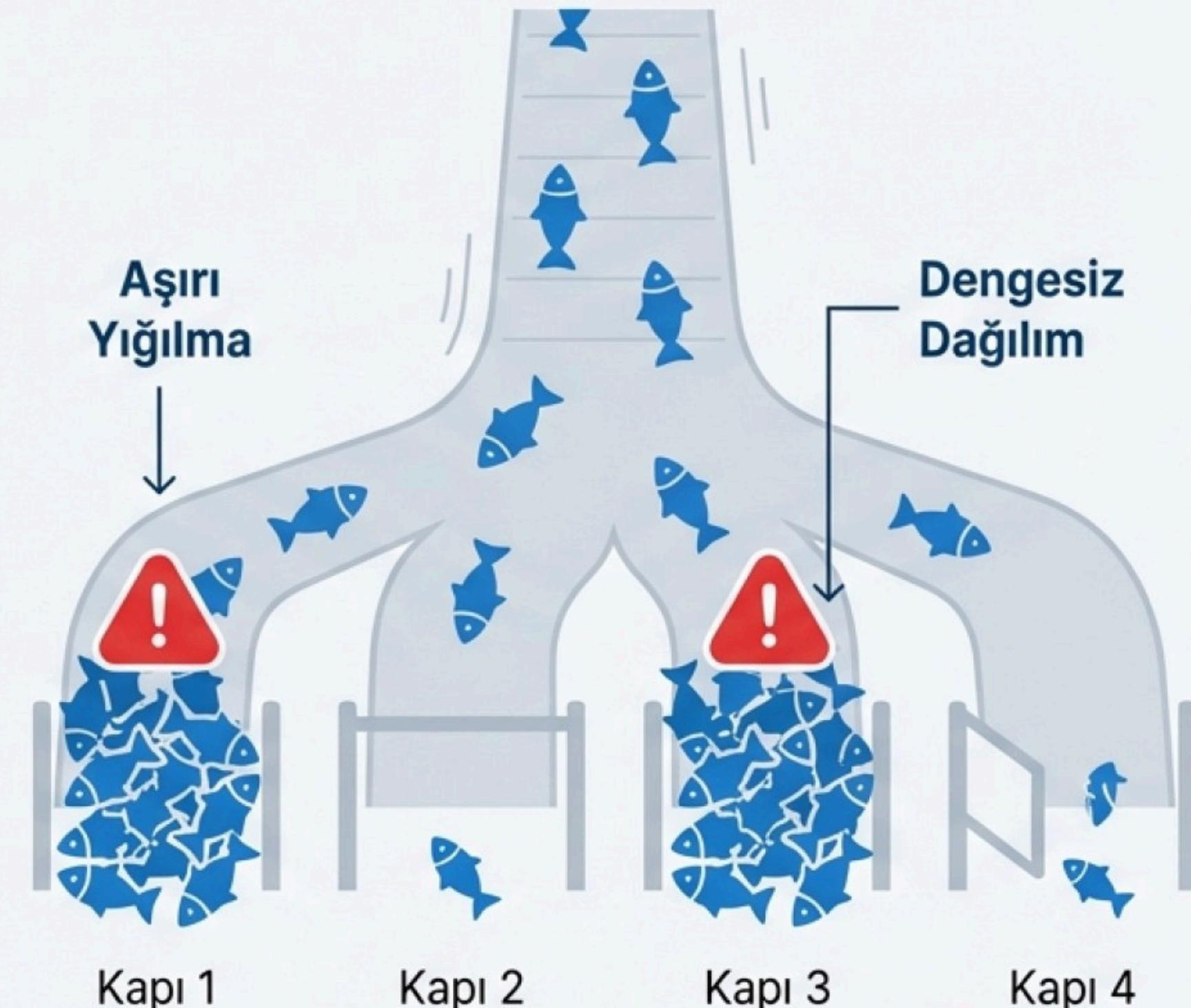
Bazı kapılarda aşırı yiğilma yaşanır.

- **Yanlış Atamalar**

Balıklar yanlış ağırlık sınıflarına yönlendirilir.

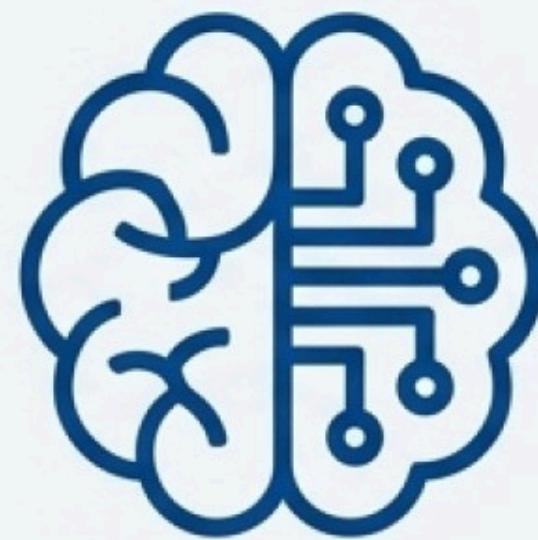
- **Artan Yeniden İşleme**

Hatalı sıralama nedeniyle ek manuel müdahale ve maliyet ortaya çıkar.



Proje Amacı: Daha Akıllı, Daha Hızlı ve Kendi Kendine Öğrenen Bir Sistem

Mevcut sensör verilerini (giriş/çıkış zamanları, anlık ağırlık, per-fish step counter vb.) kullanarak, gelen her balık için doğru kapıyı gerçek zamanlı olarak tahmin edecek ve böylece sistemin genel verimliliğini artıracak bir sınıflandırıcı geliştirmek.



Akıllı



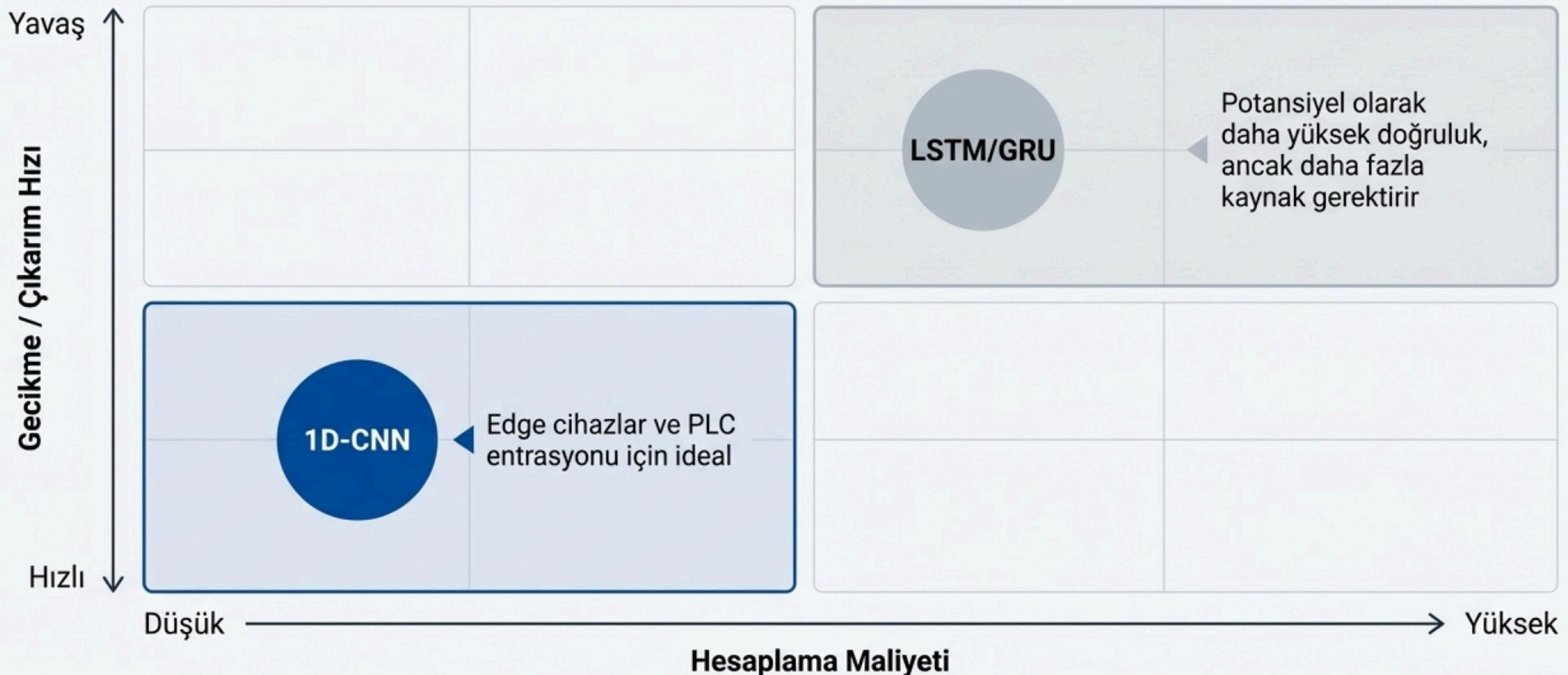
Hızlı



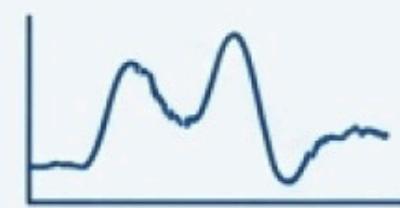
Verimli

Neden 1D-CNN? Hız ve Entegrasyon için Stratejik Bir Mimari Seçimi

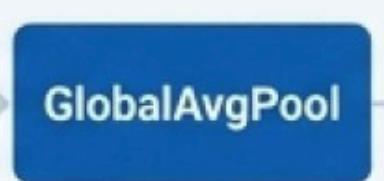
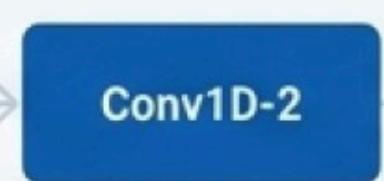
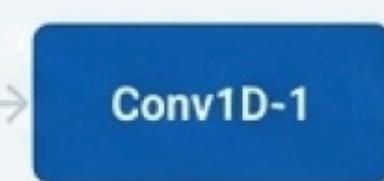
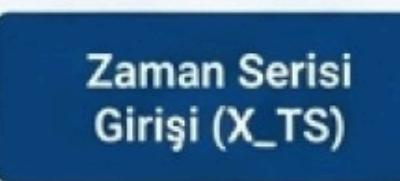
Bu projenin en kritik gereksinimi 'düşük gecikme' ve 'PLC entrasyonu'dur.



İki Veri Akışını İşleyen Hibrit Model Mimarisi



Weight profile signal

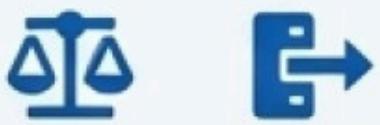


Zaman Serisi Giriş (X_TS)

Yerel zaman serisi örüntülerini yakalar

Daha karmaşık örüntülerini yakalar

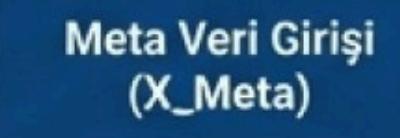
Özellik vektörüne dönüştürür



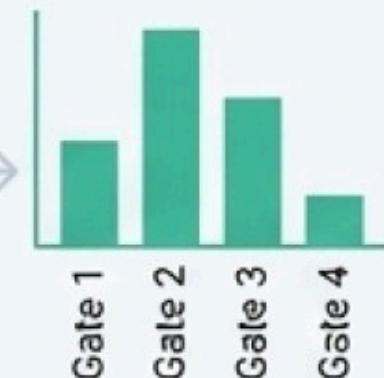
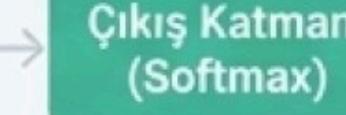
Scale ID



Step Counter



Meta Veri Giriş (X_Meta)



Gate 1

Gate 2

Gate 3

Gate 4

Özellik vektörlerini birleştirir

Modelin Katman Yapısı ve Kod Anatomisi

Modelin Ana Katmanları

Katman Tipi	Parametreler	Fonksiyon
TS Giriş	(T, 1)	Ağırlık Profilini sisteme alır.
Conv1D-1	F=16, K=5, ReLU	Yerel zaman serisi örüntülerini yakalar.
Conv1D-2	F=32, K=3, ReLU	Daha karmaşık örüntüleri yakalar.
GlobalAvgPool	---	CNN çıktısını sabit uzunlukta özellik vektörüne dönüştürür.
Meta Giriş	(K)	Ek bağlamsal özellikleri alır.
Concatenate	---	İki özellik vektörünü birleştirir.
Dense Başlık	64, ReLU	Birleştirilmiş özellikler üzerinde sınıflandırma yapar.
Çıkış Katmanı	4, Softmax	Her Gate için olasılık tahmini yapar.

Keras Kod Anatomisi

```
import tensorflow as tf
from tensorflow.keras.layers import Input, Conv1D, GlobalAveragePooling1D, Dense, Concatenate
from tensorflow.keras.models import Model

# İki girişin tanımı
input_ts = Input(shape=(T, 1), name='input_ts')
input_meta = Input(shape=(K,), name='input_meta')

# f_cnn (Zaman serisi özelliklerinin çıkarımı)
x = Conv1D(filters=16, kernel_size=5, activation='relu', name='conv1d_1')(input_ts)
x = Conv1D(filters=32, kernel_size=3, activation='relu', name='conv1d_2')(x)
f_cnn = GlobalAveragePooling1D(name='global_avg_pool')(x)

# İki kolun birleştirilmesi
t_combined = Concatenate(name='concatenate')([f_cnn, input_meta])

# Son sınıflandırma katmanı
x = Dense(64, activation='relu', name='dense_baslik')(t_combined)
output = Dense(4, activation='softmax', name='cikis_katmani')(x)

# Modeli oluştur
model = Model(inputs=[input_ts, input_meta], outputs=output)
```

Hipotez Doğrulama: Gerçek Veriye Geçmeden Önce Sistemi Test Etmek



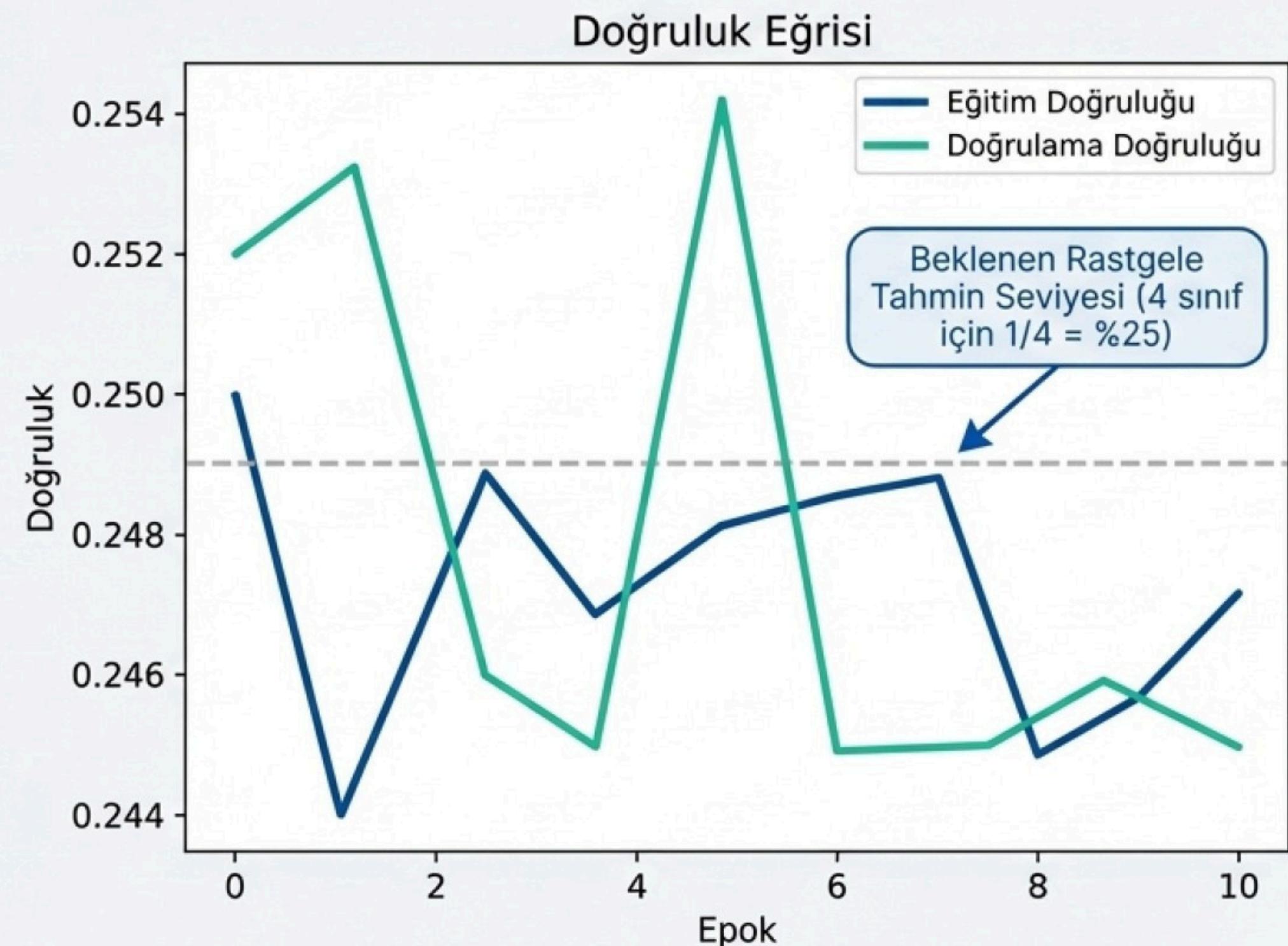
Modelin, çalışma mantığını ve veri şekillerini doğrulamak amacıyla 50.000 adet **varsayımsal** (rastgele) veri örneği üzerinde test edildiği açıklanır.

Bu testin amacı yüksek doğruluk elde etmek değil, aksine modelin rastgele veride 'öğrenmemesi' gerektiğini kanıtlamaktır.

Bu, metodolojik sağlamlığın bir göstergesidir.

1D-CNN Modeli Rastgele Veride Beklenen Davranışı Sergiledi

Eğitim sürecinde Doğrulama Kaybının (Validation Loss) ≈ 1.4 ve Doğrulama Doğruluğunun (Validation Accuracy) ≈ 0.25 seviyesinde sabit kaldığı belirtilir. Bu durumun, modelin gürültüden anlamlı olmayan desenler öğrenmediğini ve sistemin doğru çalıştığını gösterdiği açıklanır.



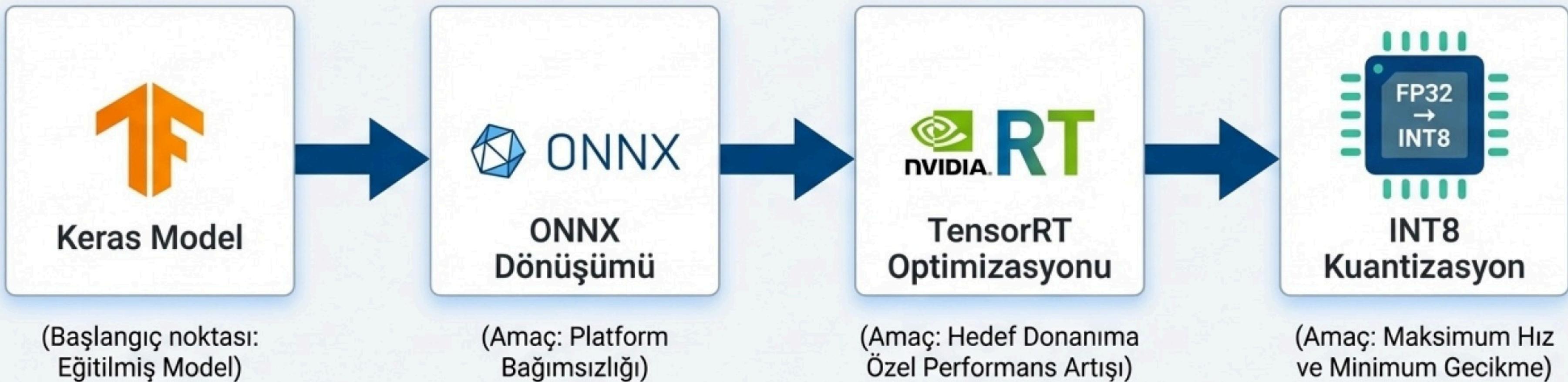
Sektör Standardı Baseline (LightGBM) Bulguları Teyit Ediyor

Kıyaslama modeli olarak, özellik mühendisliği yapılmış verilerle eğitilen LightGBM kullanılmıştır. LightGBM'in de 1D-CNN gibi %25 civarında doğruluk elde etmesi, verilerin rastgele olduğunu teyit etmiş ve her iki modelin de bekendiği gibi çalıştığını vurgulamıştır.

Metrik	LightGBM Sonucu	Beklenen Sonuç (Rastgele)	Sonuç
Test Doğruluğu	0.2587	~0.25	Teyit Edildi
F1-Skorları	≈0.26	~0.25	Teyit Edildi

****Analiz Notu**: Bu sonuç, gerçek veride 1D-CNN'in zaman serisi özelliklerini otomatik yakalama yeteneği sayesinde LightGBM'den daha yüksek performans sergileyeceği hipotezini destekler.**

Edge Cihaz Üzerinde Gerçek Zamanlı Çalışma için Optimizasyon Yol Haritası



Bu adımlar, modelin PLC entegrasyonu için kritik olan milisaniye seviyesindeki çıkışım gereksinimlerini karşılamasını sağlayacaktır.

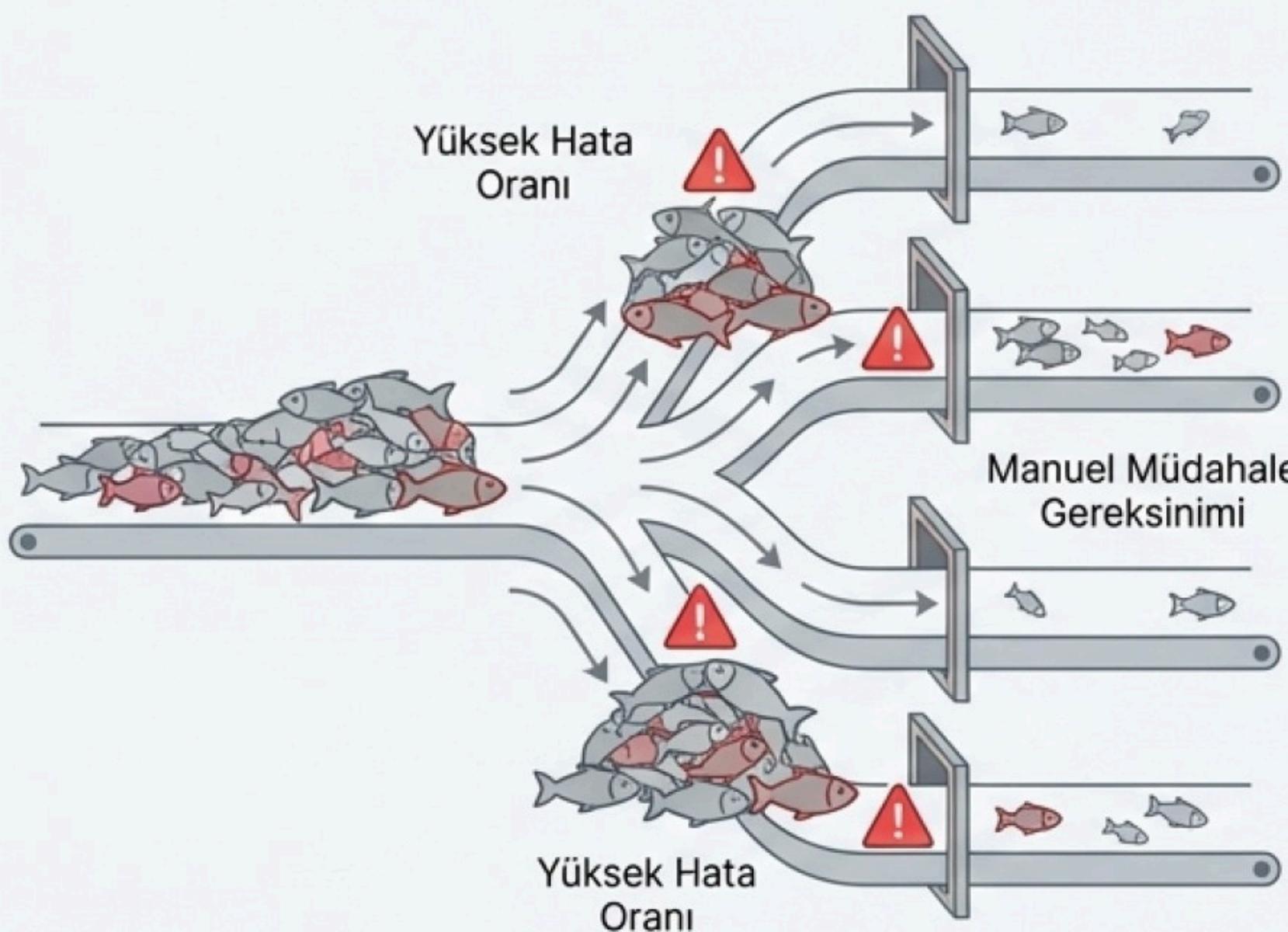
Sürekli İyileştirme Fırsatları: Doğruluk ve Hız Dengesini Optimize Etmek

Gelecekte, doğruluğun kritik olduğu senaryolar için 1D-CNN modelinin, potansiyel olarak daha yüksek doğruluk sunan LSTM/Transformer tabanlı modellerle kıyaslanacağı belirtilir. Bu çalışmanın amacının, projenin ihtiyaçları için en uygun “doğruluk ↔ gecikme” dengesini bulmak olduğu vurgulanır.

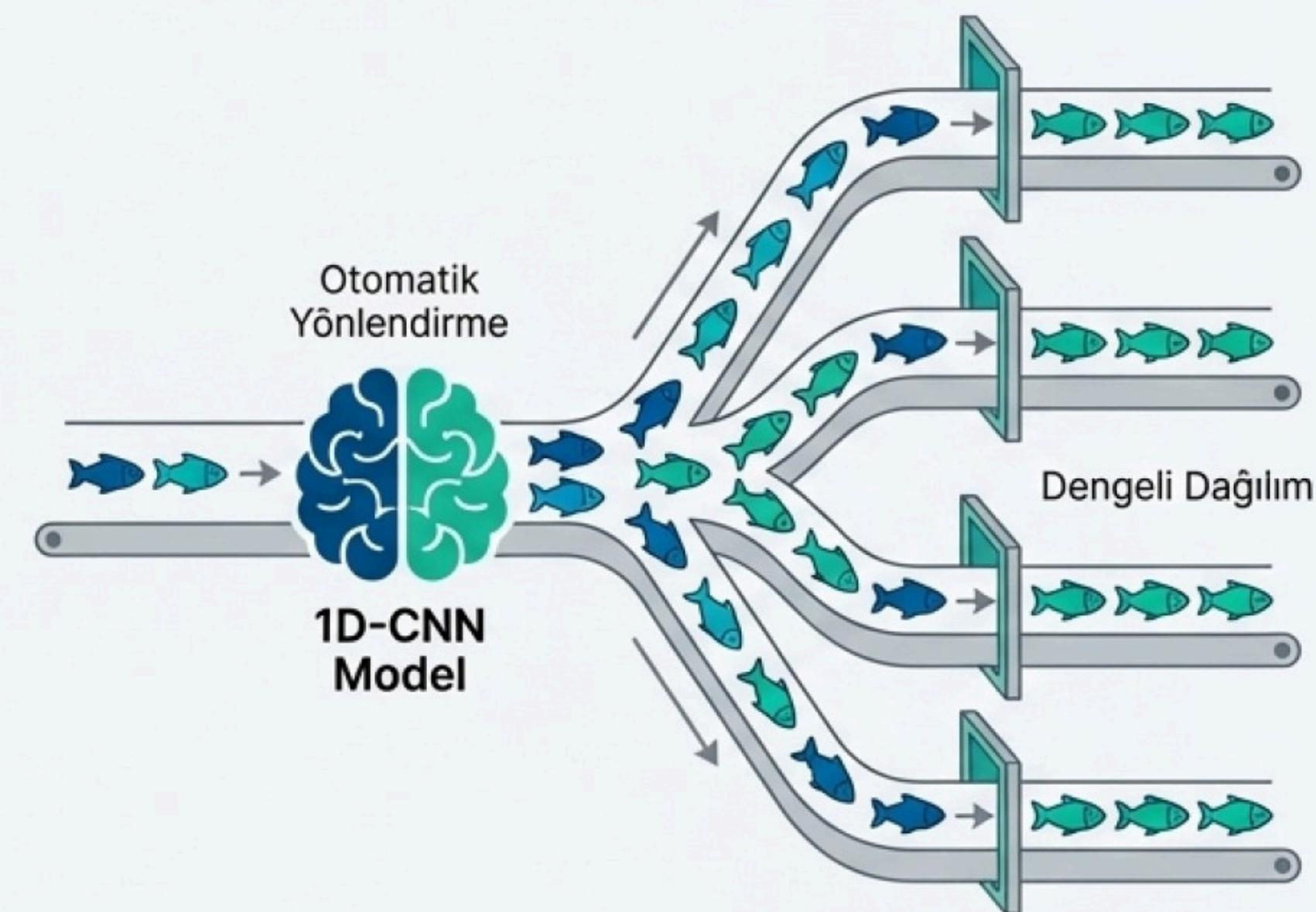


Manuel Sınırlardan Akıllı Otomasyona: Kanıtlanmış Bir Dönüşüm Yolu

Mevcut Durum: Statik ve Verimsiz



Hedeflenen Durum: Dinamik ve Verimli



Geliştirdiğimiz model, sistem verimliliğini artırmak ve operasyonel maliyetleri düşürmek için net bir yol haritası sunmaktadır.